CORRECTED VERSION*

PCT

WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H04L 12/56, 29/14 | A3

(11) International Publication Number: WO 99/21322

(43) International Publication Date: 29 April 1999 (29.04.99)

(21) International Application Number: PCT/US98/21984

(22) International Filing Date: 16 October 1998 (16.10.98)

(30) Priority Data:

60/062,581	20 October 1997 (20.10.97)	US
60/062,984	21 October 1997 (21.10.97)	US
09/059,896	14 April 1998 (14.04.98)	US

(71) Applicant: THE FOXBORO COMPANY [US/US]; 33 Commercial Street B52-1J, Foxboro, MA 02035 (US).

(72) Inventors: HIRST, Michael, D.; 146 Howland Road, Lakeville, MA 02347 (US). GALE, Alan, A.; 22 Leonard Street, Carver, MA 02330 (US). CUMMINGS, Gene, A.; 95 Old Orchard Road, Sherborn, MA 01770 (US).

(74) Agent: POWSNER, David, J.; Choate, Hall & Stewart, Exchange Place, 53 State Street, Boston, MA 02109 (US).

(81) Designated States: AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, DE, DK, EE, ES, FI, GB, GE, GH, GM, HR, HU, ID, IL, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published

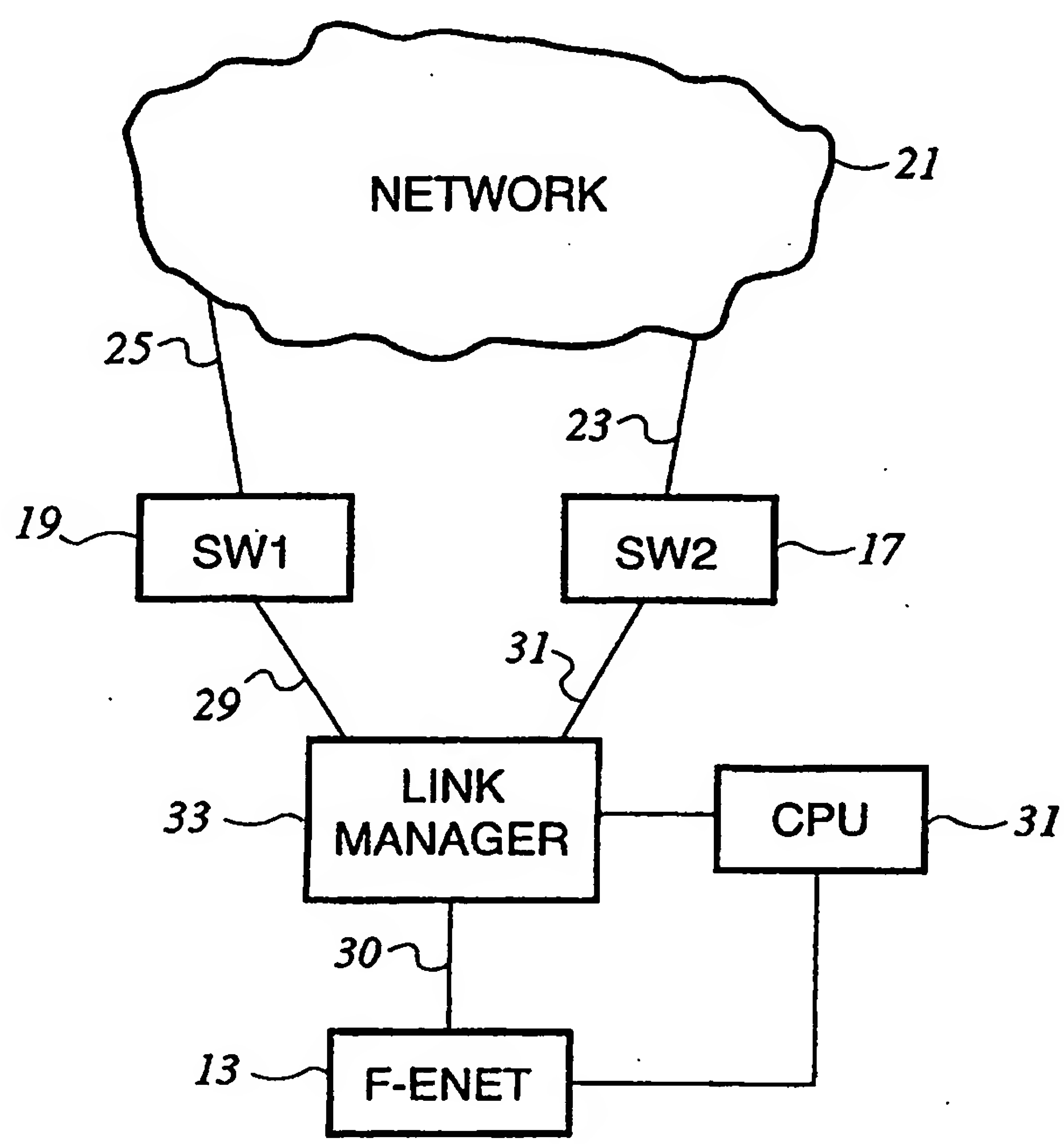
With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.

(88) Date of publication of the international search report: 19 August 1999 (19.08.99)

(54) Title: METHOD AND SYSTEM FOR FAULT-TOLERANT NETWORK CONNECTION SWITCHOVER

(57) Abstract

A computer is connected to redundant network switches by primary and secondary connections, respectively. Test messages are sent across each connection to the attached switches. A break in a connection, or a faulty connection, is detected upon a failed response to one of the test messages. In response to this failure, traffic is routed across the remaining good connection. To facilitate fast protocol rerouting, a test message is sent across the now active connection bound for the switch connected to the failed connection. This message therefore traverses both switches causing each to learn the new routing. Rerouting is therefore accomplished quickly.



*(Referred to in PCT Gazette No. 40/1999, Section II)

METHOD AND SYSTEM FOR FAULT-TOLERANT NETWORK CONNECTION SWITCHOVER

Cross Referencing To Related Patents

This patent application is related to co-pending patent applications: "Fast Re-Mapping For Fault Tolerant Connections" Serial Number 60/062681, Filed: 10/20/97; and "Fast Re-Mapping For Fault Tolerant Connections", Serial Number 60/062984, Filed: 10/21/97 both of which are incorporated by reference herein in their entireties.

Technical Field

The present invention relates, in general, to fault-tolerant computing. More specifically, the present invention relates to methods and systems for quickly switching between network connections.

Background of the Invention

The reliability of computer based applications continues to be an important consideration. Moreover, the distribution of applications across multiple computers, connected by a network, only complicates overall system reliability issues. One critical concern is the reliability of the network connecting the multiple computers. Accordingly, fault-tolerant networks have emerged as a solution to insure computer connection reliability.

In many applications, the connection between a single computer and a network is a critical point of failure. That is, often a computer is connected to a network by a single physical connection. Thus, if that connection were to break, all connectivity to and from the particular computer would be lost. Multiple connections from a single computer to a network have therefore been implemented, but not without problems.

Turning to Fig. 1, a diagram of a computer 11 connected to a network 21 is shown. Computer 11 includes a network interface, for example, a fast-Ethernet interface 13. A connection 30 links fast-Ethernet interface 13 with a fault-tolerant transceiver 15. Fault tolerant transceiver 15 establishes a connection between connection 30 and one of two connections 29 and 31 to respective fast-Ethernet switches 19 and 17 (these "switches" as used herein are SNMP managed network Switches). Switches 17 and 19 are connected in a fault-tolerant matter to network 21 through connections 23 and 25.

Fault-tolerant transceiver 15 may be purchased from a number of vendors including, for example, a Digi brand, model MIL-240TX redundant port selector; while fast-Ethernet switches 17 and 19 may also be purchased from a number of vendors and may include, for example, a Cisco brand, model 5000 series fast-Ethernet switch.

Operationally, traffic normally passes from fast-Ethernet interface 13 through fault-tolerant transceiver 15, and over a primary connection 29 or 31 to respective switch 17 or 19 and on to network 21. The other of connections 29 and 31 remains inactive. Network 21 and switches 17 and 19 maintain routing information that directs traffic bound for computer 11 through the above-described primary route.

In the event of a network connection failure, fault-tolerant transceiver 15 will switch traffic to the other of connection 29 and 31. For example, if the primary connection was 31, and connection 31 broke, fault-tolerant transceiver 15 would switch traffic to connection 29.

When, for example, traffic from computer 11 begins passing over its new, backup connection 29 through switch 19, network routing has to be reconstructed such that traffic bound for computer 11 is routed by the network to the port on switch 19 that connection

29 is attached to. Previously, the routing directed this traffic to the port on switch 17 that connection 31 was attached to.

Several problems arise from the above-described operation. First, the rebuilding of network routing to accommodate passing traffic over the back-up connection may take an extended period of time. This time may range from seconds to minutes, depending upon factors including network equipment design and where the fault occurs. Second, fault-tolerant transceiver 15 is only sensitive to a loss of the physical receive signal on the wire pair from the switches (e.g., 17 and 19) to the transceivers. It is not sensitive to a break in the separate wire pair from the transceiver to the switch. Also, it is sensitive only to the signal from the switch to which it is directly attached and does not test the backup link for latent failures which would prevent a successful recovery. This technique also fails to test the switches themselves.

Another example of a previous technique for connecting a computer 11 to a network 21 is shown in Fig. 2. Network switches 17 and 19 and their connection to each other and network 21 is similar to that shown in Fig. 1. However, in this configuration, each of switches (e.g., 17 and 19) connects to its own fast-Ethernet interface (e.g., 13 and 14) within computer 11.

Operationally, only one of interfaces 13 and 14 is maintained active at any time. When physical signal is lost to the active interface, use of the interface with the failed connection is ceased, and connectivity begins through the other, backup interface. The backup interface assumes the addressing of the primary interface and begins communications. Unfortunately, this technique shares the same deficiencies with that depicted in Fig. 1. Rerouting can take an extended period of time, and the only failure

mode that may be detected is that of a hard, physical connection failure from the switch to the transceiver.

The present invention is directed toward solutions to the above-identified problems.

Summary of the Invention

Briefly summarized, in a first aspect, the present invention includes a method for managing network routing in a system including a first node, a second node and a third node. The first node has primary and secondary connections to the second and third nodes, respectively. Also, the second and third nodes are connection by a network.

The method includes periodically communicating between the first and the second or third node over at least the primary connection. A status of network connectivity between the communicating nodes is thereby determined.

If the network connectivity determined is unacceptable, roles of the primary and secondary connections are swapped to establish new primary and secondary connections. A message is then sent with an origin address of the first node to the second node over the new primary connection. The origin address of this message facilitates the network nodes learning about routing to the first node over the new primary connection.

As an enhancement, the first node may include a first port connected to the primary connection and a second port connected to the secondary connection. The first and second ports have first and second network addresses, respectively; and the first node has a system network address. The periodic communication may be transmitted from the first port of the first node with an origin address of the first port. Further, the origin address of the message sent if network connectivity was unacceptable may be the system network address of the first node. Also, the periodic communication may be a ping

message having the first network address of the first port as its origin address. This ping message may be destined for the second or third node.

If the ping message fails, another ping message may be sent from the second port to the other of the second and third nodes, not previously pinged. If this ping message is successful, the method may include swapping the roles of the primary and secondary connections and pinging the second node over the new primary link.

As yet another enhancement, the status of the connection between the second port and the other of the second and third nodes to which the previous ping was sent is determined.

In another aspect, the present invention includes a system for implementing methods corresponding to those described hereandabove. In this embodiment a link manager may be attached to the computer and may provide connectivity between the computer and the primary and secondary connections. As implementation options, the link manager may be, for example, integral with the computer (e.g., on a main board of the computer), on an expansion board of the computer, or external to the computer. Also, the computer may be an operator workstation or a controller such as, for example, an industrial or environmental controller.

Brief Description of the Drawings

The subject matter regarded as the present invention is particularly pointed out and distinctly claimed in the concluding portion of the specification. The invention, however, both as to organization and method of practice, together with further objects and advantages thereof, may best be understood by reference to the following detailed description taken in conjunction with the accompanying drawings in which:

Figs. 1-2 depict prior art systems for managing fault-tolerant network connections;

Fig. 3 depicts a fault-tolerant network connection topology in accordance with one embodiment of the present invention;

Fig. 4 is a functional block diagram of the link manager of Fig. 3 in accordance with one embodiment of the present invention;

Figs. 5-7 are flow-diagrams of techniques in accordance with one embodiment of the present invention; and

Figs. 8-11 depict several topologies in conformance with the techniques of the present invention.

Detailed Description of a Preferred Embodiment

In accordance wit the present invention, depicted herein are techniques for establishing a fault-tolerant connection to a network that overcome the disadvantages of prior techniques discussed hereinabove. That is, according to the present invention, connectivity problems are quickly detected, and upon assumption of an alternate (back-up) connection, network reroute times are mitigated.

Turning to Fig. 3, a fast-Ethernet interface 13 is connected to both a link manger 33 and a CPU 31. The topological relationship between fast-Ethernet interface 13, link manager 33 and CPU 31 will vary with implementation requirements. Several example topologies are discussed hereinabove in regard to Figs. 9-12; however, many other topologies will become apparent to those of ordinary skill in the art in view of the disclosure herein.

The techniques disclosed herein are not limited to fast-Ethernet technology. Other networking technologies may be subjected to the techniques disclosed herein, such as, for example, conventional Ethernet technology.

Link manager 33 is connected to both fast-Ethernet interface 13 and CPU 31. The connection to fast-Ethernet interface 13 is that which would be normally used for network connectivity. The connection of link manager 33 to CPU 31 is for configuration and control purposes. In one implementation example, link manager 33 and fast-Ethernet interface 13 may each be PCI cards within a personal computer architecture. In this example. their connections to CUP 31 are by way of the PCI bus. A cable may connect fast-Ethernet interface 13 and link manager 33.

Two network connections 29 and 31 (for example, fast-Ethernet connections) couple link manager 33 to switches 19 and 17, respectively. Connections 23 and 25 couple switches 17 and 19 to network 21, which connects them to each other.

Link manager 33 is more specifically depicted in Fig. 4. A fast-Ethernet interface 41 provides connectively (e.g., PCI bus interface) with an attached host computer. Computer interface 45 also attaches to the host computer and facilitates configuration and control of link manager 33. Fast-Ethernet interfaces 47 and 49 provide redundant network connectivity. Lastly, logic 43 interconnects the above-described elements. In a preferred embodiment, logic 43 is implemented as an ASIC; however, the particular implementation of logic 43 will vary with product requirements. In other implementation examples, logic 43 could be implemented using a programmed processor, a field programmable gate array, or any other form of logic that may be configured to perform the tasks disclosed therefor herein.

To briefly summarize, the techniques of the present invention send test messages across each connection of the link manger to the attached switches. A break in a connection, or faulty connection, is detected upon a failed response to one of the test messages. In response to this failure, traffic is routed across the remaining good connection. To facilitate fast protocol rerouting, a test message is sent across the now active connection bound for the switch connected to the inactive connection. This message traverses both switches causing each to learn the new routing. Rerouting is therefore accomplished quickly.

More particularly, according to one-embodiment, Figs. 5-6 depict flow-diagrams of operational techniques in accordance with one embodiment the present invention. To begin, the link manger pings a switch connected to the primary, active connection, every T_p seconds, STEP 101. The ping message contains a source address unique to the link manager port currently associated with the active connection. If the active connection is ok, pinging thereof continues, STEP 101. Also, a check is regularly performed to detect a loss of receive signal on the active connection interface, STEP 113.

If either pinging fails on the active connection, or carrier has been lost, a test is performed to check whether the back-up connection status good, STEP 105. If the back-up connection is unavailable, no further action can be taken and pinging of the primary connection continues in anticipation of either restoration of the active connection or availability of the back-up connection. Also under this condition, the host computer may be notified such that it may take appropriate action, such as, e.g., to enter a fail-safe condition.

If the back-up connection status is good, the link manger is configured to direct traffic through the back-up connection, STEP 107. Further, a ping message is sent from

the link manger, through the switch connected to the back-up connection and to the switch connected to the primary, failed, connection, STEP 109. This ping message contains a source address of the computer connected to the link manager. As a result, the switches connected to the primary and back-up connections are made aware of the new routing to the computer. This facilitates the immediate routing of traffic bound for the computer over the back-up, secondary, connection. Lastly, the roles of active and backup connections are swapped and the process iterates, STEP 111.

Turning to Fig. 6, a flow-diagram depicts a technique for maintaining the status of the back-up connection. A ping is send over the back-up connection to its respective switch every T_p seconds, STEP 115. The ping message contains a source address unique to the link manager port currently associated with the backup connection. If the back-up connection is good, that is, the ping is responded to timely, STEP 117; then the back-up connection status is set to good, STEP 119. If the response to the ping message is not timely received, the back-up connection status is set to bad, STEP 121. (A maintenance alert may also be generated. The invention facilitates detecting latent faults in unused paths and repairing them within the MTBF of a primary fault.) In either case, the processor iterates to the pinging step, STEP 115.

According to the above-described embodiments ping messages are sent from the link manager, across each connection to the switch attached thereto. Failure of these ping messages will indicate failure of the link the ping message was sent across. In accordance with the embodiment of Fig. 7 described below, ping messages are sent across each link, but are bound for the switch connected to the other connection. Thus, the ping message must traverse one switch to get to the destination switch, traversing both the connection from the link manager to the immediately attached switch and across the connection

between the switches. Thus, the technique described below can localize faults in the connections between the link manager and each switch and the connection between the switches. Further, this embodiment contains example information on how time message transmission can be implemented using a common clock.

As described above, the pings sent from each port have unique source address for that particular port. However, to facilitate fast rerouting, the final ping, once the port roles are swapped uses the source address of the attached computer system.

To begin, a clock tick is awaited, STEP 201. Clock ticks are used as the basis for timing operations described herein. If a clock tick has not occurred, no action is taken. However, if a clock tick has occurred a first counter is decremented, STEP 203. This first counter is designed to expire, on a 0.5 second basis (of course, this time can be adjusted for particular application requirements).

If the first counter expired, indicating that the 0.5 second period has elapsed, a ping message is sent from the active port to the standby switch using the address of the active port, STEPS 205, 207. If the ping is successful, STEP 209, a second counter with a 30 second interval is decremented, STEP 211. The second counter decrement is also performed if the first counter decrement did not result in the 0.5 second time period expiring, STEP 205. If the second counter has not expired, STEP 213, then the process iterates awaiting a next clock tick, STEP 201. If the second counter has expired, a ping is sent from the standby port to the active switch using the standby port's address, STEP 215. If the ping was successful, STEP 217 then the process iterates awaiting another clock tick, STEP 201.

If the ping from the active port to the standby switch failed, STEP 209, a ping is sent from the standby port to the active switch, STEP 227. If this ping is successful, STEP

229, then the roles of the active and standby ports and switches are reversed, STEP 231, and a ping is sent from the now active port to the now standby switch using the address of the computer station, STEP 233. This ping facilitates the switches learning the new path to the computer thus correcting routing information. Furthermore, the old active port is determined to be in error, STEP 235.

Turning back to STEP 215, if the ping from the standby port to the active switch failed (STEP 217) a ping is sent from the active port to the standby switch, STEP 219. If this ping fails, there is an error associated with the standby port, STEP 223.

Turning back to STEP 227, a ping was sent from the standby port to the active switch. If this ping failed, then the current error must be associated with either the switches, the network between the switches or both ports may be bad. Therefore, for the following steps, it is most helpful to refer to the ports and switches as the "A port", "A switch", "B port" and "B switch", wherein the A port is directly connected to the A switch and B port is directly connected to the B switch. The notion of which port is currently active and which port is currently backup is not significant to the following steps.

Again, if the ping from the standby port to the active switch, STEPS 227, 229, failed then a ping is sent from the A port to the A switch, STEP 237. If this ping is successful, STEP 239, then the A port is set as the active port, STEP 241. A ping is then sent from the B port to the B switch, STEP 243. If this ping failed STEP 245, then the error is associated with B switch, STEP 247; however, if the ping was successful, then the error is associated with the network, STEP 249.

If the ping from the A port to the A switch, STEP 237, failed, STEP 239, then the B port is set as active, STEP 251. A ping is then sent from the B port to the B switch, STEP 253. If this ping failed, then an error is associated with both ports, STEP 259;

however, if the ping was successful, STEP 255, then the error is associated with A switch, STEP 257.

In each of the above steps, once the error is determined and set (STEPS 223, 235, 247, 249, 257, and 259), an interrupt is sent to the host processor (STEP 255) for providing notification of the change in network configuration.

The techniques of the present invention may be implemented in different topologies. As examples, several of these topologies are depicted in Figs. 8-11.

In each of the examples, the computer depicted may be, for example, a workstation, an embedded processor, a controller, (e.g., industrial or environmental) or other computer type.

Beginning with Fig. 8, a computer 11 is depicted and contains fast-Ethernet interface 13 and link manager 33 connected by cable 30. Connections 29 and 31 couple the system to a network. The particular implementation and use of computer 11 will vary. In one example, computer 11 is a PCI bus-based computer and fast Ethernet interface 13 and link manager 33 are PCI interface cards. In another embodiment, all circuitry may be on a common board (e.g., the system motherboard).

In Fig. 9, the functions of link manager 33 and fast-Ethernet interface 13 have been integrated onto a single interface card. As one example, this card may interface with its host computer using a PCI bus.

In Fig. 10, fast-Ethernet interface 13 is incorporated on a main board (e.g., a motherboard) of computer 11. Link manager 33 is a peripheral (e.g., PCI) interface card.

In Fig. 11, fast-Ethernet interface 13 may be incorporated on a main board of computer 11 or as a separate interface card. Link manager 33 is disposed external to computer 11 and is connected thereto by connections 30 and 63. Connection 63 is

particularly used for command and control of link manager 33 and interfaces with computer 11 through a communications port 61 (e.g., a serial or parallel port).

A variety of techniques are available for implementing the techniques described herein. The present invention is not meant to be limitive of such implementation, as many options are available to those of ordinary skill in the art and will be apparent in view of the disclosure herein. Implementations may take form of software, hardware, and combinations of both. Dedicated logic, programmable logic, and programmable processors may be used in the implementation of techniques disclosed herein. One particular implementation example using programmable logic to implement a simple instruction set capable of implementing the techniques described herein is described in detail in Appendix A, "HDS 5608-Dual Switched Ethernet Interface, Revision 1.1" attached hereto and incorporated by reference herein in its entirety.

While the invention has been described in detail herein, in accordance with certain preferred embodiments thereof, many modifications and changes thereto can be affected by those skilled in the art. Accordingly, is intended by the appended claims to cover all such modifications and changes as fall within the true spirit and scope of the invention.

****
"PRELIMINARY, SUBJECT TO CHANGE WITHOUT NOTICE, DO NOT USE FOR PRODUCTION".

PRELIMINARY

Draft #9,2/18/97

HDS 5608
Dual Switched Ethernet Interface
Revision 1.1
G. Cummings

COMPUTER INTERFACE TITLE: "Computer Interface Title Here"

KEY WORDS:
"Key words here"

RELATED DOCUMENTS:

PSD C01E: Platform Enhancements for High Performance and High Reliability Control Market

CPS 5591: High Performance and High Reliability Data Acquisition Control Market Hardware/Software Modifications

CONFIDENTIAL
For Specifically Authorized Personnel Only
DO NOT PHOTOCOPY

HDS 5608

TABLE OF CONTENTS

1. Design Objectives . . . . . -18-
   1.1 Design Description . . . . . -18-
   1.2 Design Purpose . . . . . -19-
   1.3 Design Objectives . . . . . -19-
   1.4 Interrelationship to System . . . . . -19-

2. Reference Documents . . . . . -20-

3. Functional Specifications . . . . . -20-
   3.1 Internal Hardware-Oriented Functions . . . . . -20-
   3.2 Firmware Description . . . . . -21-
   3.3 Diagnostics . . . . . -21-

4. Principle of Operation . . . . . -21-
   4.1 Architecture . . . . . -21-
      4.1.1 The Link Manager . . . . . -22-
      4.1.2 The Satellite Receiver Time Strobe . . . . . -29-
      4.1.3 The Foxboro Letterbugs . . . . . -30-
   4.2 Bus Descriptions . . . . . -31-
      4.2.1 Electrical Characteristics . . . . . -32-
      4.2.2 Data Movement . . . . . -32-
      4.2.3 Constraints . . . . . -32-
      4.2.4 Programming Information . . . . . -32-
   4.3 Interfaces . . . . . -33-
   4.4 Technology Applications and Constraints . . . . . -33-
   4.5 Functional Block Description . . . . . -33-
   4.6 Testability/Fault Isolation Features . . . . . -33-

5. Hardware Oriented Performance . . . . . -33-
   5.1 Performance Requirements . . . . . -33-
   5.2 Performance Goals . . . . . -34-
   5.3 Constraints . . . . . -34-
   5.4 Cycle Time/Bit Rates/Speed . . . . . -34-
   5.5 Power Requirements . . . . . -34-
   5.6 FMEA Results . . . . . -34-

6. Special Design Considerations . . . . . -34-
   6.1 Power/Grounding Constraints . . . . . -34-
   6.2 Packaging . . . . . -35-
   6.3 Physical Constraints/Implications . . . . . -35-

6.4 *Environmental Constraints/Limitations* . . . . -35-
6.5 *Product Safety/Certification Considerations* . . . . -35-
6.6 *Test Considerations* . . . . -35-

HDS 5608

REVISION HISTORY FOR HDS 5608

REVISION	1.0	DECEMBER 1997	
REVISION	1.1	FEBRUARY 1998	DEBUG FEATURES ADDED INCLUDING REGISTER REORGANIZATION FOR DUMPING TO MEMORY.

HDS 5608

1. Design Objectives

The Dual Switched Ethernet Interface (DSEI) circuit board is intended to provide an interface between any PC or workstation having standard PCI bus slots and the High Performance I/A network. This requires three different circuits which are unrelated except for the common PCI interface.

The major function on the board is the Link Manager, which is circuitry for achieving communications redundancy between individual I/A stations and the first pair of switches in a network of redundant Ethernet/Fast Ethernet switches. This includes the path from the stations through any hubs present and from the hubs to the switches. Above the first pair of switches the fault tolerance of the network is a function of the system purchased.

Another new function is the interface to the Satellite Receiver Time Strobe. This is an optional feature which allows stations in the I/A High Performance Network to synchronize their real time clocks.

The board also contains the standard I/A Letterbug Interface, necessary to establish the identity of each station on the network.

1.1 Design Description

The cable redundancy function of the DSEI accepts one Ethernet/Fast Ethernet MII (Media Independent Interface) from the host station and switches it between two Ethernet/Fast Ethernet PHY chips on the DSEI. These drive two cables to separate switches in the network which should have no common mode of failure between them. The DSEI contains a programmable Link Manager which selects one of the two PHY driver chips and its link to the network as active and the other as standby. It monitors the operability of each by sensing its link integrity signal and by periodically sending heartbeat messages to the first level switches of the network to which it is connected and monitoring the reply. If it finds either link to be inoperable it reports the failure for maintenance, and if the failure is on the active link it can be programmed to automatically switch to the standby.

For time synchronization the station first receives a message giving the time at which the strobe will occur. The Time Strobe is a simple pulse which interrupts the processor so it can set its real time clock to the time given in the preceding message. The station may be set up as a master, which receives the message and strobe from the satellite receiver and distributes them to other stations, or as a slave which simply receives the strobe and interrupts the processor. The strobe also resets a counter which counts milliseconds from the strobe for a software readable high resolution elapsed time reference.

The Letterbug interface is the same as is used on present I/A modules. It consists of the input and output ports necessary to read the unique hard-wired links within each of the six letterbugs plugged into the board. This allows software to identify the module identifier characters they represent.

HDS 5608

1.2 Design Purpose

The chief purpose of the design is to maintain the same level of station level cable redundancy and fault recovery as the present I/A network while upgrading network performance with commercially available Fast Ethernet Switches which do not inherently support such fast switching redundant station connections. It also provides hardware support for the Time Strobe and letterbug functions, which are unique to I/A, to the commercial PC or workstation in which it is used.

1.3 Design Objectives

The objective of the DSEI board is to integrate the station in which it is used into the I/A High Performance Network and maintain a communications failure recovery time of no more than one second between peer group stations.

1.4 Interrelationship to System

The same Link Manager circuit will be used in each station having redundant network interfaces. In each case it will connect to the MII standard interface of the host's Fast Ethernet controller and its output will consist of two RJ-45 connectors for the A and B ports which connect to the network. A red Category 5 UTP cable will be used to connect the A port to the A switch of a redundant pair, and a similar green cable will be used to connect the B port to the B switch of the pair. Initialization and control of the Link Manager is accomplished by the host over a standard PCI interface.

For use with commercial processors the DSEI circuit will be packaged on a standard PCI board occupying one slot. It will connect to a Fast Ethernet controller either on the motherboard or on a separate PCI card using a standard MII cable and connector. To further integrate these commercial processors into the I/A system, this board will also contain a standard I/A letterbug and interface and an interface for receiving and optionally retransmitting an I/A time sync strobe.

For use with Foxboro processors the circuit will be packaged on the main processor board where it will connect directly to the PCI bus and the MII of the Fast Ethernet controller. For Z Modules the two network connections will be made through the module I/O connector. The letterbug and time sync strobe will be located elsewhere on the processor motherboard.

Where optical fiber connections to the network are required they will be supplied by external wire-to-fiber converters in the case of commercial processors or by a fiber uplink from the associated hub in the case of Foxboro modules. In either case the device must supply the link integrity signal.

The same circuitry is intended to be used in all future products requiring redundant connections to the switched Ethernet/Fast Ethernet network with some variations in packing similar to those described above.

HDS 5608

The Time Strobe will generally be received from the satellite receiver by a master station which will distribute it to all other stations in the installation requiring it via a daisy chained shielded twisted-pair cable and RS-485 transceivers. See HDS 5624 for details.

2. Reference Documents

PSD C01E: Platform Enhancements for High Performance and High Reliability Data Acquisition and Control Market.

CPS 5591: High Performance and High Reliability Data Acquisition and Control Market Hardware/Software Modifications.

HDS 5563: Fast Ethernet Control Processor Modules.

HDS 5624: Hardware for Computer Time Synchronization.

HDS 1017: Module Identifiers. (Letterbugs)

PCI Local Bus Specification Rev 2.1, PCI Special Interest Group, June 1995.

New Products Catalog, PLX Technology, July 1997.

Data Sheet LXT970, Fast Ethernet Transceiver, Level One Communications, Inc. Rev 1.1, May 1997.

3. Functional Specifications

3.1 Internal Hardware-Oriented Functions

HDS 5608

The Link Manager hardware consists of a programmable logic chip and a 32K X 8 bit Link Manager memory used as a program and message buffer. The programmable chip is controlled by commands from the host and any faults detected are signaled by interrupt and presented to it in a status register.

When commanded to perform message operations it uses program code and data from the Link Manager memory which is mapped into PCI memory space and must be downloaded prior to such operations. This contains a set of "canned" messages to be transmitted, and a control program for using them to determine the health of the network interface and to maintain communications within the peer group.

When running, the Link Manager chip controls access to its own registers and to the Link Manager memory in order to prevent interference from the processor with its operations. It must be halted by issuing an I/O command, with execution verified in the Status Register, before the Link Manager memory can be accessed.

The physical ports on the board are called Port A and Port B. These physical ports are assigned a logical role, active or standby, by the Link Manager.

3.2 Firmware Description

EEProms will be used to automatically configure the internal registers of the PCI 9050-1 and the Altera EPF6016 at power up. The Link Manager algorithm and data will be downloaded to the Link Manager memory by the host processor. See the Architecture section for a description of the Link Manager algorithm.

3.3 Diagnostics

Since the DSEI is to be used with commercial PC's and workstations there will be no start up diagnostics. Manufacturing diagnostics will still be required to check out the first boards before a GenRad fixture becomes available.

4. Principle of Operation

4.1 Architecture

The DSEI card for use with commercial workstations and file servers includes three unrelated functions. The major one is the Link Manager for maintaining dual fault tolerant connections to the network. The card also contains the Foxboro letterbug module identifiers and the Time Strobe Interface for transmitting and receiving accurate time signals throughout the network.

HDS 5608

4.1.1 The Link Manager

The Link Manager is a special purpose processor which serves as an agent of the communications software since the latter does not have the low level link and physical level control to execute the required functions continuously in real time. The program and data required are downloaded to the Link Manager at system configuration time and it acts autonomously to maintain an operational link to the network thereafter.

4.1.1.1 Memory and register addressing

Registers and commands use Local Address Space 0 of the PLX chip, which should be configured as 16 I/O locations based at PCI Base Address 0. Programs and stored messages are contained in a 32Kx8 memory chip which is addressed as Local Address Space 1 of the PLX chip and configured as a 32K memory space based at PCI Base Address 1.

In addition to the host addressing above there are also 4-bit Local Register Addresses (LREG 3..0) used by the Link Manager program. Some registers are only accessible from the host, some only by the Link Manager, and some by both.

4.1.1.2 Registers

Local address space 0 (I/O)
Address Offset = 0H
Write Only, data is don't care

Halt Command

7	6	5	4	3	2	1	0

Halts instruction execution and places link manager into halt mode, necessary to download to the Link Manager memory or registers. Halted condition must be verified by reading Status Register before downloading.

Local address space 0 (I/O)
Address Offset = 1H
Read/Write if Halted

Control Register

7	6	5	4	3	2	1	0
RN	SS	MD 10	AP	CTL		IMSK	

RN Run state. Setting starts continuous program execution at address in IP.

HDS 5608

SS When set with run, initiates single step execution of one instruction at IP address. At completion, loads contents of internal registers to high end of memory to aid debugging, clears run and halts.
MD10 Changes communication clocks to 10MHz for standard Ethernet. Used in association with software initiated auto-negotiation.
AP Assigned port for operational communications. 0 = A port, 1 = B port.
CTL Additional control bits if required for future use.
IMSK Interrupt mask bits. 0 enables related interrupt, 1 disables related interrupt. IMSK0 = Link Manager, IMSK1 = Time Strobe.

Host Local Address Space 0 (I/O)
Address = 2H, Read Only
LREG = 2H

Status Register

7	6	5	4	3	2	1	0
LA	LB	E5	E4	E3	E2	E1	E0

LA = 1 Failure of Link Integrity signal on A port.
LB = 1 Failure of Link Integrity signal on B port.
E5 - E0 Error code used to define network fault to host.

Local Address Space 0 (I/O)
Address = 3H, Read Only
LREG = 3H

Interrupt Register

7	6	5	4	3	2	1	0
						TS	LM

Both bits are set to interrupt host and are cleared by host when read.

LM = 1 Link Manager Interrupt.
TS = 1 Time Strobe Interrupt

Local Address Space 0 (I/O)
Address = 5H, Read/Write if Halted
LREG = Not Accessible

Instruction Pointer Low Byte

7	6	5	4	3	2	1	0
IP 7..0							

Local Address Space 0 (I/O)
Address = 6H, Read/Write if Halted
LREG = Not Accessible

HDS 5608

Instruction Pointer High Byte

7	6	5	4	3	2	1	0
	IP 14..8						

Not accessible from host
LREG = OH
Read/Write

Real Time Clock Register

7	6	5	4	3	2	1	0
F7	F6	F5	F4	F3	F2	F1	F0

This register contains the value used to automatically reload the counter which counts a 41.9 43 ms pulse (the period of 25 MHz/2^20) from the hardware to provide the real time clock tick. It pulses TICK for 1us upon reaching zero to initiate each pass of the scheduler and to decrement timeout counters. For example, a register value of 05H produces a TICK period of approximately 200 ms. The value should be set by a compromise between the shortest counted period, the ping timeouts, and the longest counted period, the standby channel ping rate.

Not accessible from host
LREG = 1H
Read/Write

Flag Register

7	6	5	4	3	2	1	0
	TO	TI	F4	F3	F2	F1	F0

Z Zero flag set by operations producing a zero result in the accumulator.
TO Set by a timeout result of the ping instruction.
TI Set by countdown to zero of the real time clock register (tick). Must be reset by programmed logic instruction.
F4 - F0 Programmable flags which can be used for program flow logic.

4.1.1.3 Interrupts

The PLX 9050-1 chip maps all interrupts from the DSEI board to the PCI bus interrupt for the slot in which it is located. It defines two maskable hardware interrupts plus a software interrupt and the Interrupt Control/Status Register contains enable, status and polarity control bits for each, plus a master enable. Owing to an error in the initial chip which makes these difficult to differentiate in operation however, only the default hardware

HDS 5608

interrupt is used, Local Interrupt 1. It should be configured for active high operation and enabled at power up.

The Control and Interrupt Registers contain separate mask and interrupt bits for the Link Manager and the Time Strobe. These drive Local Interrupt 1 of the PLX chip. Interrupt mask bits inhibit interrupts when set to one. Interrupt Register bits define the cause of interrupt when set and are cleared when read. The causes of Link Manager interrupt are defined by the error code in the Status Register. The only cause of the Time Strobe interrupt is receipt of the strobe.

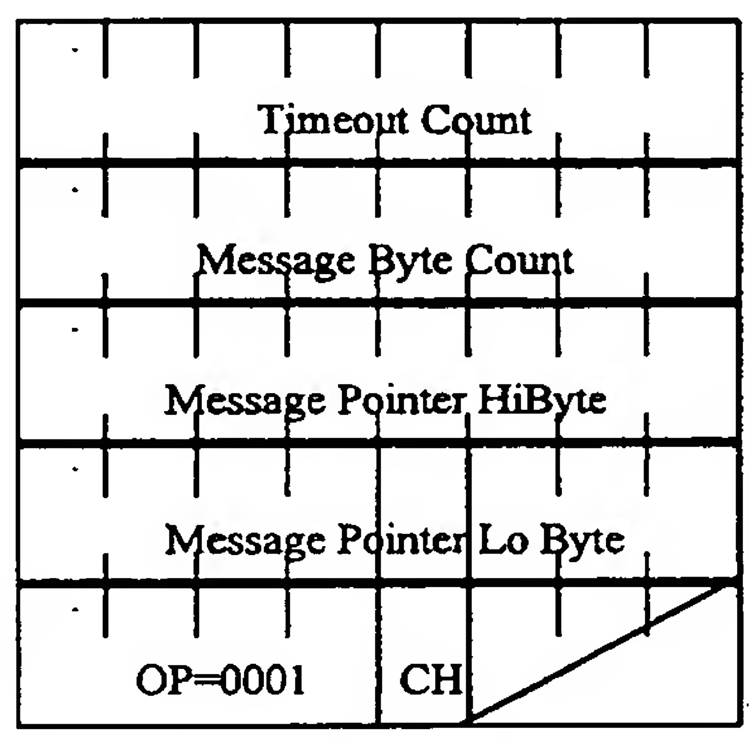
4.1.1.4 Instruction Set

The instruction set is designed to support sending stored ping messages to the network switches and testing for the reply as well as implementing real time counters for scheduling them. It also includes logical and branching operations for forming sequences of them to localize faults and make intelligent routing decisions.

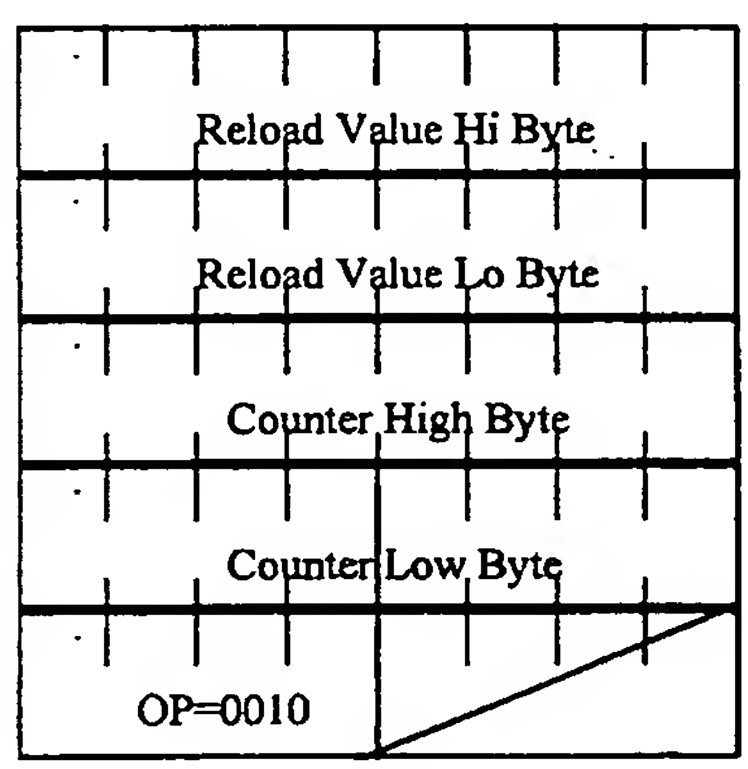
PING: Transmit ping message and test for reply. If no reply received within timeout period set TOFLAG, else clear TOFLAG.

CH = 0 Transmit ping on A chann el.

CH = 1 Transmit ping on B chann el.

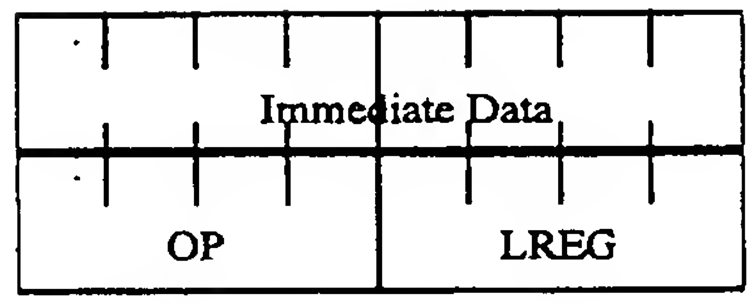


DCNT: Decrement counter. Reload counter and set ZFLAG when count reaches zero, else clear ZFLAG.



HDS 5608

LD (OP = 1111) LD register with immediate data and set/clear ZFLAG



AND (OP = 1100) AND register with immediate data, load to register and set/clear ZFLAG.

OR (OP = 1101) OR register with immediate data, load to register and set/clear ZFLAG.

XOR (OP = 1110) XOR register with immediate data, load to register and set/clear ZFLAG.

AND (OP = 0100) AND register with immediate data and set/clear ZFLAG only.

OR (OP = 0101) OR register with immediate data and set/clear ZFLAG only.

XOR (OP = 0110) XOR register with immediate data and set/clear ZFLAG only.

LREG is local Link Manager register address.

Set ZFLAG if result equals zero, else clear ZFLAG.

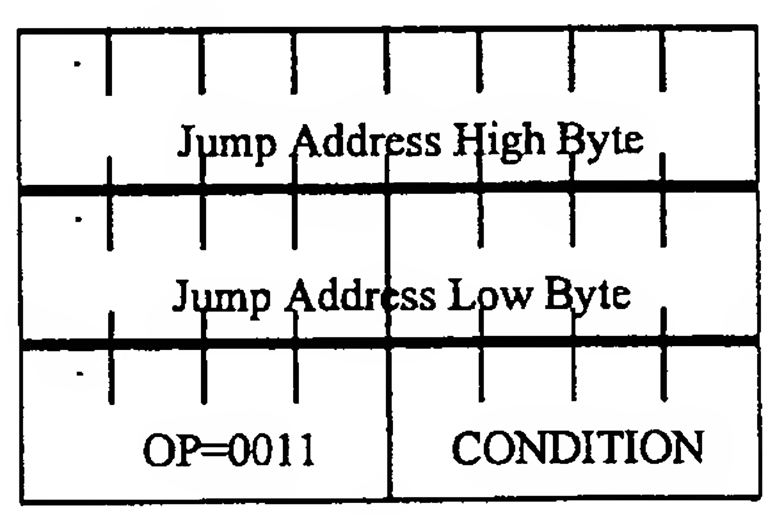
Example: 1110 0001, 0001 0000 uses an XOR operation to complement bit 4 of the Control Register, which is the means of toggling the active port assignment.

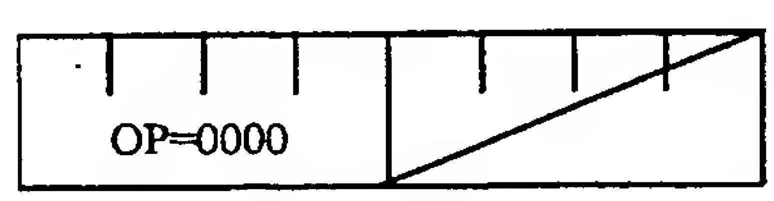
JMP: Go to jump address if condition true.

CONDITIONS

0000	Unconditional	1000	No Operation
0001	ZFLAG = 1	1001	ZFLAG = 0
0010	TOFLAG = 1	1010	TOFLAG = 0
0011	TICK = 1	1011	TICK = 0

0100-0111 and 1100-1111 are reserved.



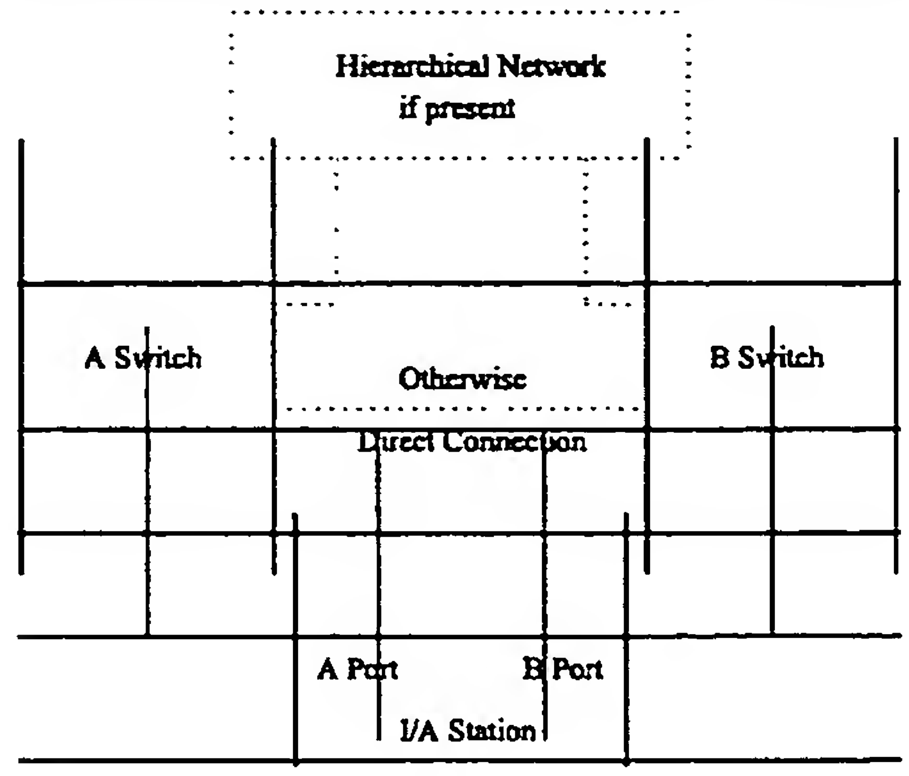


HALT: Halt instruction execution.

HDS 5608

4.1.1.5 Link Manager Operation

The Link manager is intended to operate with the following network configuration.



Each station is connected to two different switches within the network for redundancy. These are interconnected with each other directly, if they are the highest level of the network, or through the higher level network if they are not. The stations on such a pair of switches constitute a process control peer group which must maintain contact with each other through any single fault condition with no more than one second of communications loss. Longer recovery times are allowed at higher levels of the network.

The Link Manager in each station normally tests its connection to both switches since link failures may force it or other stations onto either switch. Should one of the switches themselves fail however, all stations must detect this and independently switch to the survivor. Should the interconnection between the two switches fail, whether it is direct or via the network, all stations must recognize this and choose to connect to the same switch by prior agreement.

Each station has three different MAC addresses assigned, one to the station itself for operational traffic and one to each port, used by the Link Manager for sending and receiving test messages.

A simplified flow chart of the main link management algorithm follows. It starts with a scheduler loop which runs with each tick of the real time clock. With each pass of this loop the counters for each scheduled operation are decremented and when they reach zero the link tests are executed. The active channel is tested every half second. The standby channel is also tested every 30 seconds in order to catch any latent faults.

The normal link tests consist of ping messages from each port sent to the opposite switch than the one to which they are connected. This tests the station link, both switches and the path between them. Should only one of these tests fail and not the other, it implies loss of that station link since the other sources of failure are common to both tests. If it is the standby channel which failed it is simply reported to the host for maintenance.

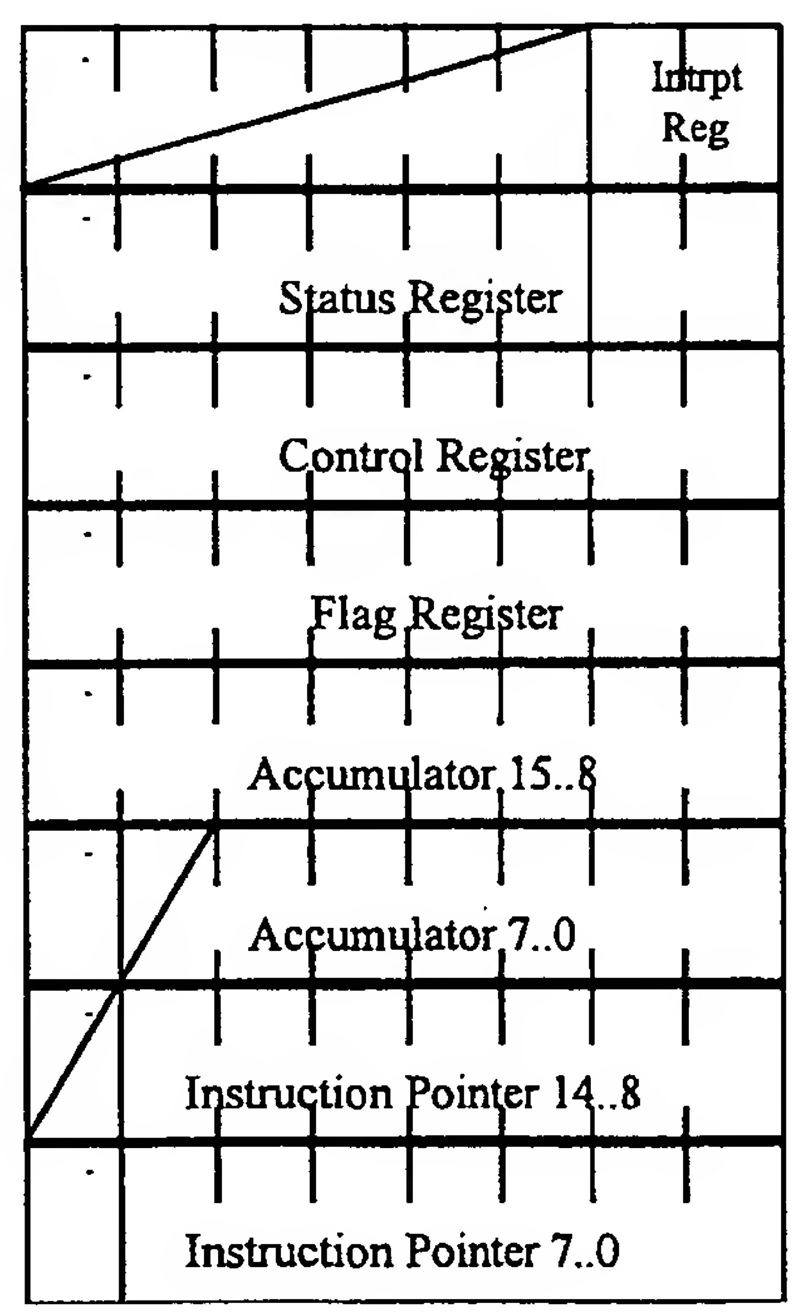
HDS 5608

Should the active channel test fail however, the roles of active and standby port are switched. A ping message using the station address is then sent to the old active switch via the new active port and switch. This causes each switch along the path to associate the station address with the port on which this message was received, which effectively reroutes all operation traffic from the old active switch to the new active switch and port onto which the station has been relocated.

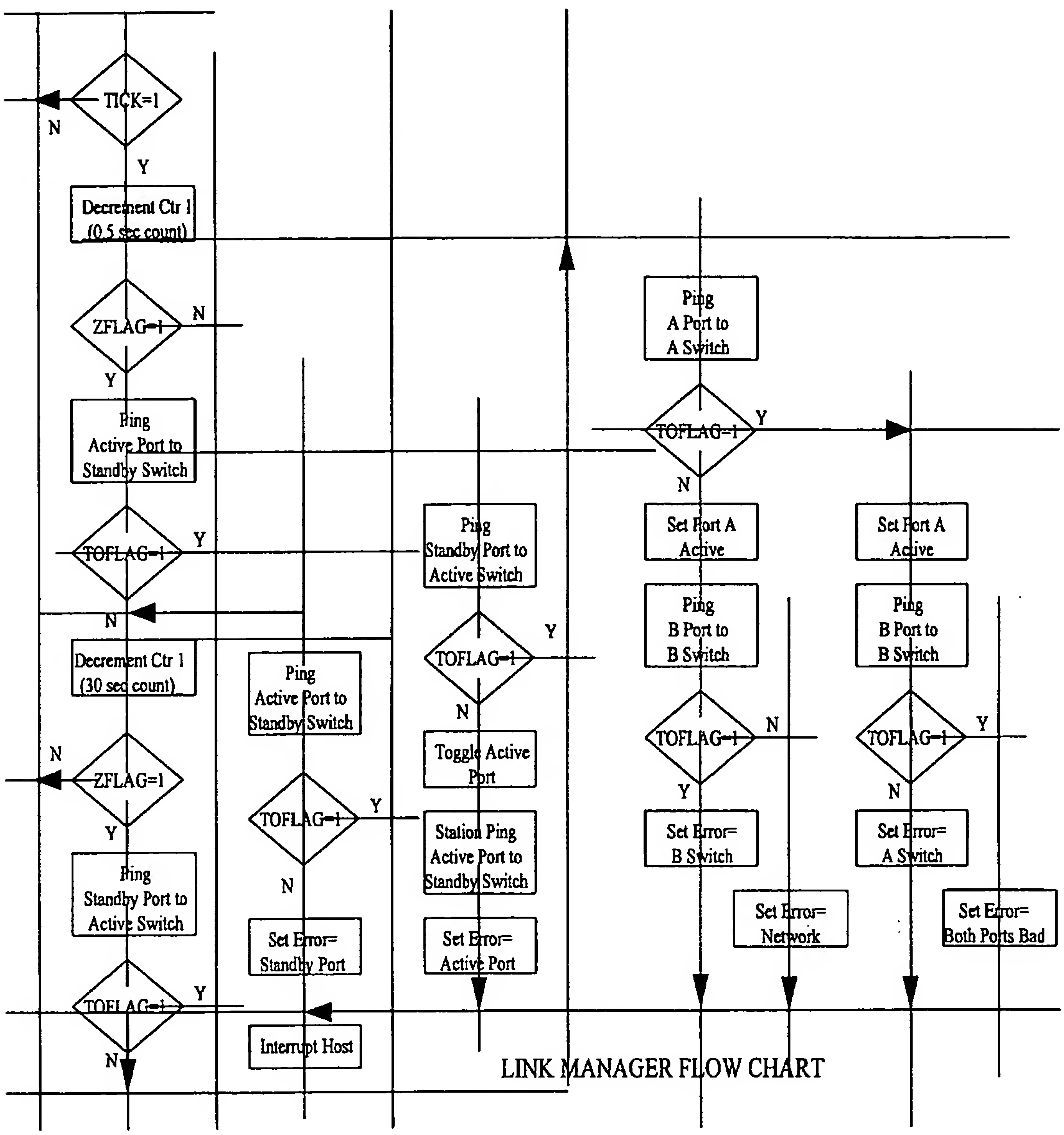
Should both tests fail, it implies that one or the other switch is bad, or the connection between them is bad. In this case each switch is tested by pinging it from the port directly connected to it. If either switch is bad, the station switches over to the good one. If both switches test good, then the connection between them is bad and all stations arbitrarily switch to the A port. If both switches should test bad, then both switches or the links to them are bad. This is a multiple fault from which no recovery by the Link Manager is possible, but the situation is reported to the host for whatever local fail safe action it can take.

4.1.1.6 Debugging Features

A single-step feature with a dump of the internal registers to the memory has been included to aid in the debugging of the Link Manager code. Following any single-step operation the contents of the following registers will be moved into the Link Manager memory locations indicated.



HDS 5608



LINK MANAGER FLOW CHART

4.1.2 The Satellite Receiver Time Strobe

The satellite receiver sends a time of day message followed by a separate Time Strobe, a single pulse marking that time by its leading edge. The message is received over separate communications. The strobe is received on the DSEI board of each station and is used to interrupt the processor.

Only one I/A station will receive these messages and pulses directly from the satellite received via an RS-232 link and strobe interrupt. It will act as the master station to distribute both of them as often as is required to the rest of the installation via the I/A network and the strobe interrupt of each station. A master station will set up its Time Strobe Control Register. All other stations must set this register to all zeroes.

Time Strobe Control Register

Local Address Space 0

Address Offset = 8H

Read/Write

7	6	5	4	3	2	1	0
DRV	ENB		SKP	C3	C2	C1	C0

DRV = 1: Drive time strobe bus directly as an output of this bit.

ENB = 1: Enable each pulse from the satellite receiver to drive the time strobe bus

SKP = 1: Enable one pulse from the satellite receiver to drive the time strobe bus after skipping the number of pulses specified in C3-C0.

For high resolution timing a readable/writeable counter is included on the board which is reset by the Time Strobe and counts out each second in millisecond increments.

High Resolution Timer High Byte

Local Address Space 0

Address Offset = 9H

Read Only

7	6	5	4	3	2	1	0
						T9	T8

HDS 5608

High Resolution Timer Low Byte
Local Address Space 0

Address Offset = AH

Read Only

7	6	5	4	3	2	1	0
7	6	5	4	3	2	EN	EN 0
·						P7	P0
T7	T6	T5	T4	T3	T2	T1	T0

4.1.3 The Foxboro Letterbugs

The Foxboro letterbugs are unique hard-wired plugs for each of the alphanumeric characters, six of which are typically inserted into each I/A station module as module identifiers. Six of these are included on the PCI version of the DSEI to identify the station in which it is used to the rest of the network.

The Foxboro letterbug interface electrically reads the six coded letterbugs. It consists of two addressable letterbug select ports enabled by two separate control bits from a third port and a read-only input port where the letterbug code may be read.

Letterbug Select Port Low Byte

Address Offset = BH

Local Address Space 0

Read/Write

7	6	5	4	3	2	1	0
·		LB 6	LB 5	LB 4	LB 3	LB 2	LB 1

Letterbug Select Port High Byte

Local Address Space 0

Address Offset = CH

Read/Write

7	6	5	4	3	2	1	0
·		LB 6	LB 5	LB 4	LB 3	LB 2	LB 1

Letterbug Pin Enbable

Local Address Space 0

HDS 5608

Address Offset = DH

Read/Write

Letterbug Code Input Port
Local Address Space 0

Address Offset = EH

Read/Write

7	6	5	4	3	2	1	0
		B5	B4	B3	B2	B2	B0

Pin 0 is driven high on a particular letterbug if it s bit in the Letterbug Select Port low byte is set and ENP0 is set to enable the pin 0 drivers. Through unique links to P0 for each character a diode array to the Letterbug Code Input Port is driven where the resulting letterbug code can be read.

Pin 7 on a particular letterbug is driven high if its bit in the Letterbug Select Port high byte is set and ENP7 is set to enable the pin 7 drivers. Through unique links to P7 for each character a diode array to the Letterbug Code Input Port is driven where the resulting letterbug code can be read.

ENP0 and ENP7 should be kept normally reset and should not both be set at the same time. Each letterbug should be selected in turn, EMP0 set, the letter bug code read, ENP0 cleared and ENP7 set, the letterbug code read again, and ENP7 cleared.

For details of the letterbug decoding scheme see Foxboro HDS 1017, "MODULE IDENTIFIERS".

4.2 Bus Descriptions

The PCI bus is converted by a PLX Technologies PCI 9050-1 chip to an 8-bit local bus with address, data and strobe signals. The programmable logic chip and the Link Manager memory chip actually reside on this local bus. The chip also provides interrupts, programmable chip selects and the PCI configuration registers for the plug-and-play interface. All registers and buffers on the DSEI are addressed by the local address space of a programmable chip select and an offset address which maps to a PCI base address with the same offset.

4.2.1 Electrical Characteristics

All bus signals comply with the relevant standards. Local bus and internal logic signals will be 5V CMOS logic levels.

HDS 5608

4.2.2 Data Movement

During initialization data will be downloaded over the 32-bit PCI bus to the 8-bit Link Manager memory and 8-bit I/O registers in the logic chip via the local bus. Byte ordering is Little Endian. During run mode messages will be read from the Link Manger memory over the 8-bit local bus and transferred via the 4-bit MII interface to the PHY interface chips. They in turn will re-encode the 4-bit values using Manchester encoding for Ethernet or 4/5 encoding for Fast Ethernet and will serialize this for transmission to the network.

4.2.3 Constraints

The network interface of the DSEI is constrained to operate half duplex since the Link Manager must be able to use the CSMA/CD media access control to gain the use of the link for sending its ping messages. This mechanism does not function in the full duplex mode of controllers since no contention for the link is possible.

Additionally, in the case of fault tolerant stations a half-duplex hub is required as the immediate station interface in order for the shadow module to receive what the primary is sending and for both to receive the same traffic from the rest of the network.

4.2.4 Programming Information

Since the DSEI board is to be used with a variety of commercial PCs and workstations it must implement the plug-and-play features of the PCI interface. The PLX 9050-1 PCI interface chip supports this.

The chip initialization information is contained in a serial EEPROM which can be initially downloaded and programmed from the host and will subsequently automatically initialize the internal registers of the chip. Part of this initialization includes setting up the PCI Configuration Registers to support the plug-and-play interface. Details are contained in the PLX manual listed under reference documents.

HDS 5608

4.3 Interfaces

The CPU Ethernet/Fast Ethernet interface to the DSEI is the Media Independent Interface (MII) specified in IEEE Standard 802.3u (Fast Ethernet).

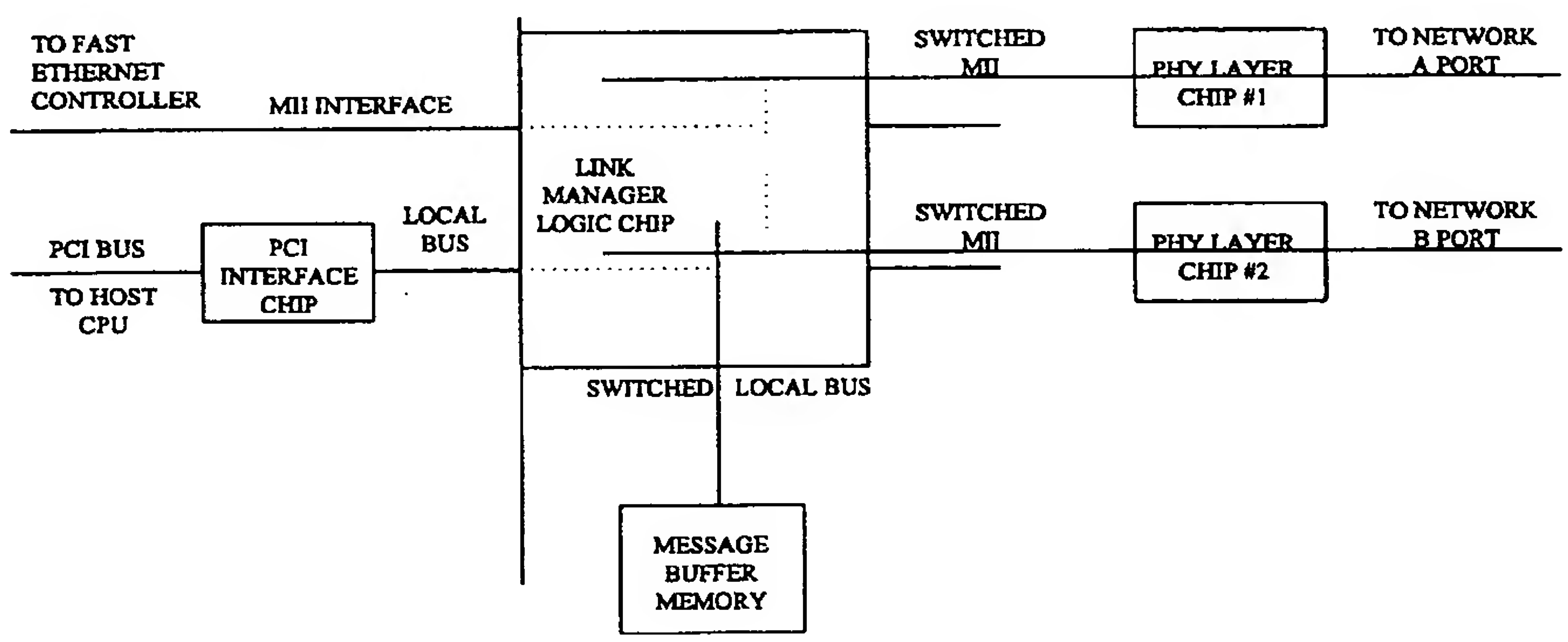
The interface from the DSEI to the network is the 10BaseT/100BaseTX auto-negotiated interface specified in IEEE Standards 802.3 and 802.3u.

The interface by which the CPU downloads and controls the DSEI is the PCI Local Bus Standard, Revision 2.1.

4.4 Technology Applications and Constraints

This interface is intended for either Ethernet or Fast Ethernet connections from the station to the network. It cannot support Gigabit Ethernet at the station level.

4.5 Functional Block Description



4.6 Testability/Fault Isolation Features

The main Link Manager algorithm performs fault isolation to the first level of switching. Since the Run command allows multiple starting locations, multiple cable maintenance and diagnostic functions can also be permanently resident in the Link Manager memory along with the normal operational code.

The Time Strobe will have an LED next to the daisy-chained cable connection which will blink with each pulse received as an aid to troubleshooting any cable problems.

5. Hardware Oriented Performance

5.1 Performance Requirements

HDS 5608

The primary performance requirement is the ability to perform fault detection and recovery within a peer group in one second or less.

HDS 5608

5.2 *Performance Goals*

The goal is to be able to switchover and reroute within a few hundred microseconds after the heartbeat timeout period.

5.3 *Constraints*

The Link Manager should operate using only industry standard functions of the network hardware and software insofar as possible in order to maintain flexibility in the choice of manufacturer. Proprietary functionality is to be avoided if possible.

5.4 *Cycle Time/Bit Rates/Speed*

The PCI Bus operates at 33 MHz. The PCI 9050-1 interface chip may insert wait states if it cannot keep up.

The MII interface operates at 2.5/25 MHz depending on whether the Ethernet or Fast Ethernet mode is in effect.

The Ethernet/Fast Ethernet serial interface operates at 10/100 MHz data rates. (Fast Ethernet actually operates at 125 MHz on the wire because of the 4/5 bit encoding.)

A common 25 MHz oscillator will clock both PHY chips and the Link Manger FPGA. The Link Manager will divide the 25 MHz Fast Ethernet clock by two to operate on bytes at 12.5 MHz except for the immediate 4-bit data interface to the PHY chips. The real time clock counter is driven by the 25 MHz divided by 2^20 to approximately 23.84 Hz.

5.5 *Power Requirements*

The DSEI requires only 5V power obtained from the PCI connector of the computer in which it is used. Power consumption is yet to be determined.

5.6 *FMEA Results*

An FMEA analysis will be done on the completed design.

6. Special Design Considerations

6.1 *Power/Grounding Constraints*

The DSEI will be laid out on a four layer printed circuit board with signals sandwiched between the power and ground planes for minimal EMC radiation. Power and ground connections to the host are made via the standard PCI connector and its pinout.

6.2 *Packaging*

HDS 5608

The DSEI is packaged as a standard full-length PCI card.

6.3 Physical Constraints/Implications

The greatest physical constraint is the space available for connectors on the metal flange of a PCI card. The letterbug and two RJ-45 connectors must be mounted there in order to be accessible from the back of the PC or workstation during operation. Therefore the permanently installed MII and Time Strobe interface cables will pass through the flange and be connected internally on the card.

6.4 Environmental Constraints/Limitations

See CPS 5591 for system requirements.

6.5 Product Safety/Certification Considerations

See CPS 5591 for system requirements.

6.6 Test Considerations

A full test of the DSEI can only be done after representative configuration of I/A High Performance network switching equipment can be made available for testing. Some test software will be required to generate continuous traffic which can be interrupted by a simulated failure. The data should contain a message count that would permit analyzing the amount of data lost during failure detection and recovery. The same software could also be used for detecting the amount of data loss caused by various simulated failures within the network itself.

Claims

We claim:

1. A method for managing network routing in a system including a first node, a second node, and a third node, wherein said first node has a primary connection to said second node and a secondary connection to said third node, wherein said node and said third node are connected by a network, and wherein said method includes:

(a) periodically communicating between said first node and one of said second node and said third node over at least said primary connection and thereby determining a status of network connectivity between said first node and said one of said second node and third node; and

(b) if said network connectivity status determined in said step (a) is unacceptable, swapping roles of said primary and said secondary connections to establish new primary and secondary connections and sending a message with an origin address of said first node to said second node over said new primary network connection, wherein said origin address of said message facilitates said network nodes learning about routing to said first node over said new primary connection.

2. The method of claim 1, wherein said first node includes a first port connected to said primary connection and a second port connected to said secondary connection, said first port having a first network address, said second port having a second network address and said first node having a system network address, wherein said periodic communication is transmitted from said first port of said first node with an original address of said first port.

3. The method of claim 2, wherein said origin address of said sending message of said step (b) comprises said system network address of said first node.

4. The method of claim 3, wherein said periodic communication between said first node and one of said second node and said third node comprises a ping message having said first network address of said first port as an origin address of said ping message.

5. The method of claim 4, wherein said ping message has a destination of said second node.

6. The method of claim 4, wherein said ping message has a destination of said third node.

7. The method of claim 4, wherein if said ping fails, a ping is sent from said second port to the other of said second node and said third node.

8. The method of claim 7, wherein if said ping from said second port to said other of said second node and said third node is successful, said method includes performing said swapping roles of said primary and secondary connections and said pinging of said second node over said new primary link of said step (c).

9. The method of claim 2, further comprising sending a ping message from said second port, with an origin address thereof, to the other of said second node and said third node to determine a status of network connectivity thereto.

10. A method for managing network routing in a system including a computer, a first network switch, and a second network switch, said first and second network switches being network connected, wherein said computer has an active connection to said first network switch and a backup connection to said second network switch, said method including:

(a) periodically pinging said second network switch by transmitting a ping message bound for said second network switch over said active connection, said ping having an address of a port of said computer connected to said active connection; and

(b) if said ping fails, and said backup connection is available, swapping roles of said active and backup connections to establish new active and backup connections and sending a ping with an origin address of said computer system to said first network switch over said new active connection, wherein said origin address of said ping facilitates said network nodes learning about routing to said computer over said new active connection, said address of said computer system being different than said address of said port.

11. A system of managing network routing including a first node, a second node, and a third node, wherein said first node has a primary connection to said second node and a secondary connection to said third node, said system including:

(a) means for periodically communicating between said first node and one of said second node and said third node over at least said primary connection and determining a status of network connectivity between said first node and said one of said second node and third node thereby;

(b) means for determining if said network connectivity status determined in said step (a) is unacceptable, and if so, for swapping roles of said primary and said secondary connections to establish new primary and secondary connections and for sending a message with an origin address of said first node to said second node over said new primary network connection, wherein said origin address of said message facilitates said network nodes learning about routing to said first node over said new primary connection.

12. The system of claim 11, wherein said first node comprises a computer.

13. The system of claim 12, further including a link manager attached to said computer, said link manager providing connectivity between said computer and said primary and secondary connections.

14. The system of claim 13, wherein said link manager is integral with said computer.

15. The system of claim 14, wherein said link manager is on a main board of said computer.

16. The system of claim 13, wherein said link manager is on an expansion board of said computer.

17. The system of claim 13, wherein said link manager is external to said computer.

18. The system of claim 12, wherein said computer comprises an operator workstation.

19. The system of claim 12, wherein said computer comprises one of an industrial controller and an environmental controller.

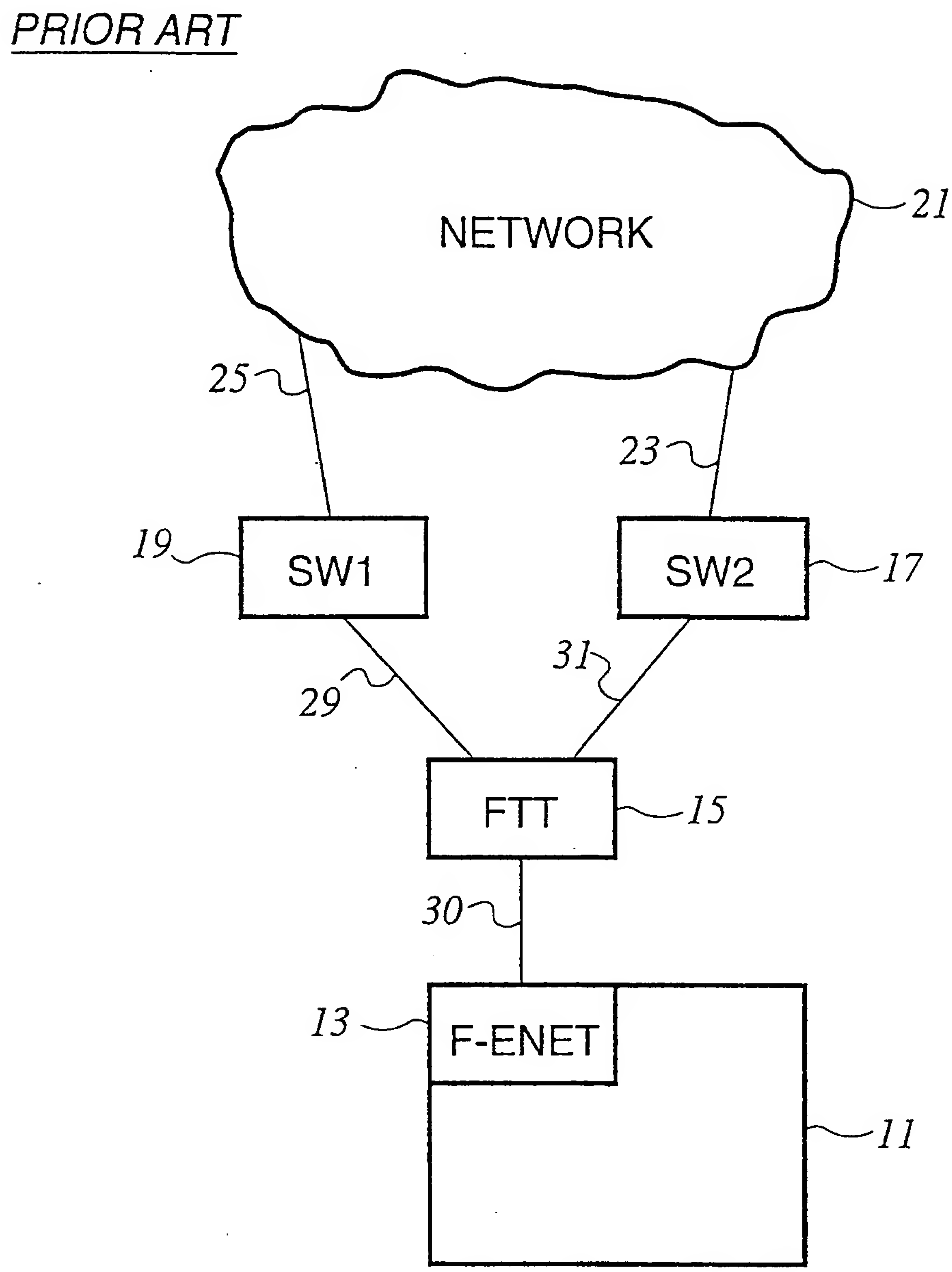

PRIOR ART
NETWORK
21
25
23
19
SW1
SW2
17
31
29
FTT
15
30
13
F-ENET
11

FIG. 1

PRIOR ART

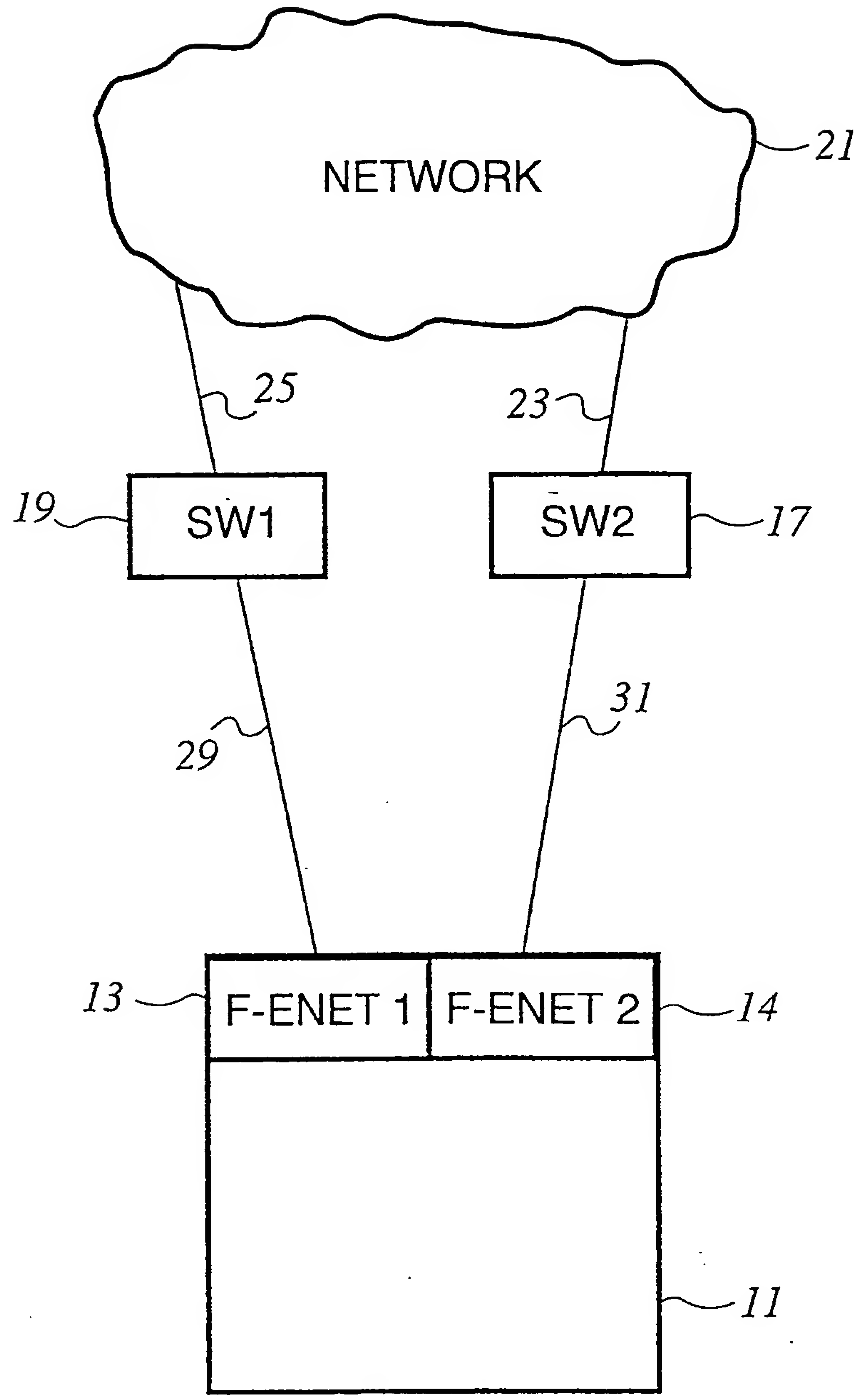


FIG. 2

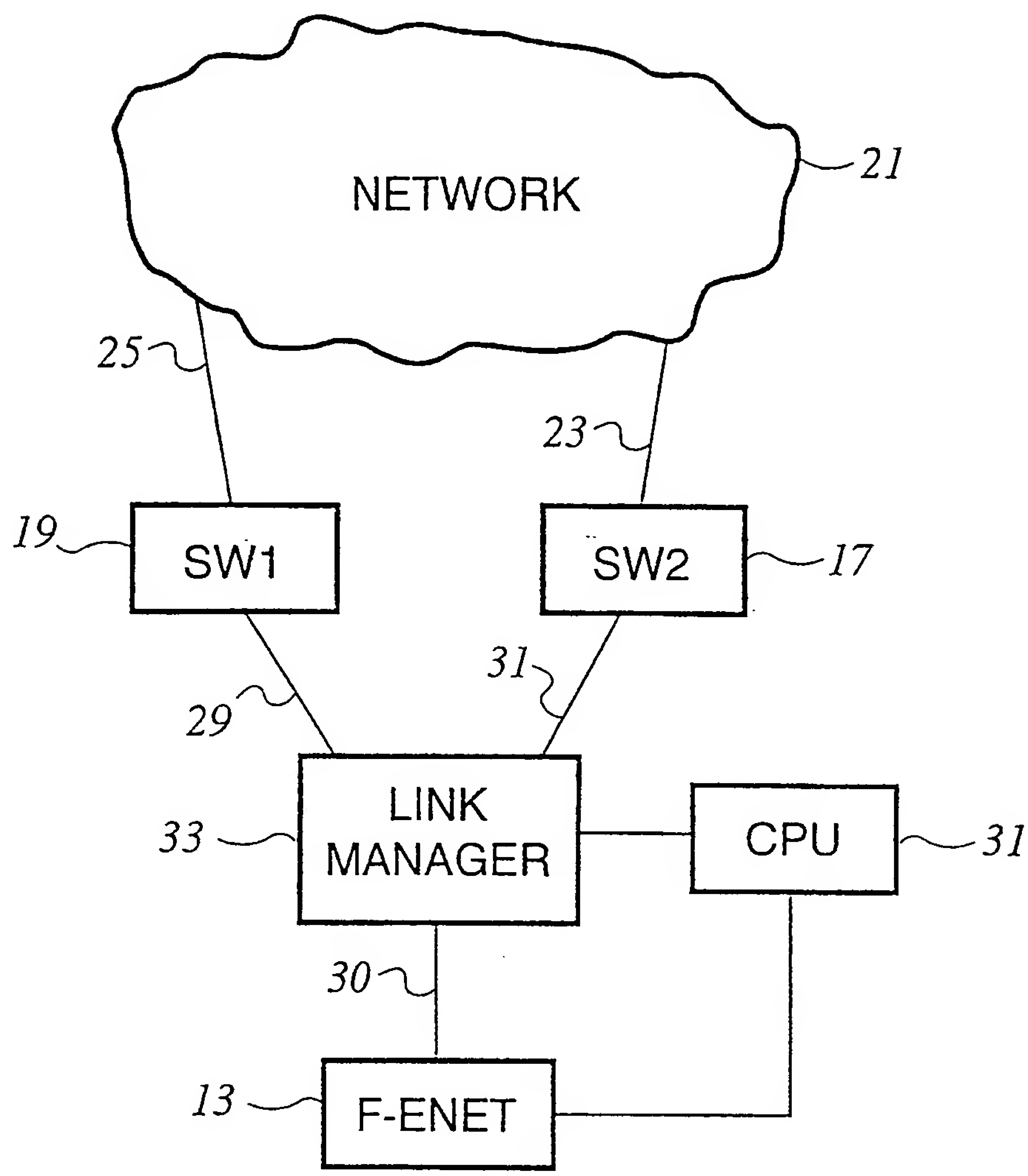

NETWORK
21
25
23
19
SW1
SW2
17
31
29
LINK
MANAGER
CPU
33
31
30
13
F-ENET

FIG. 3

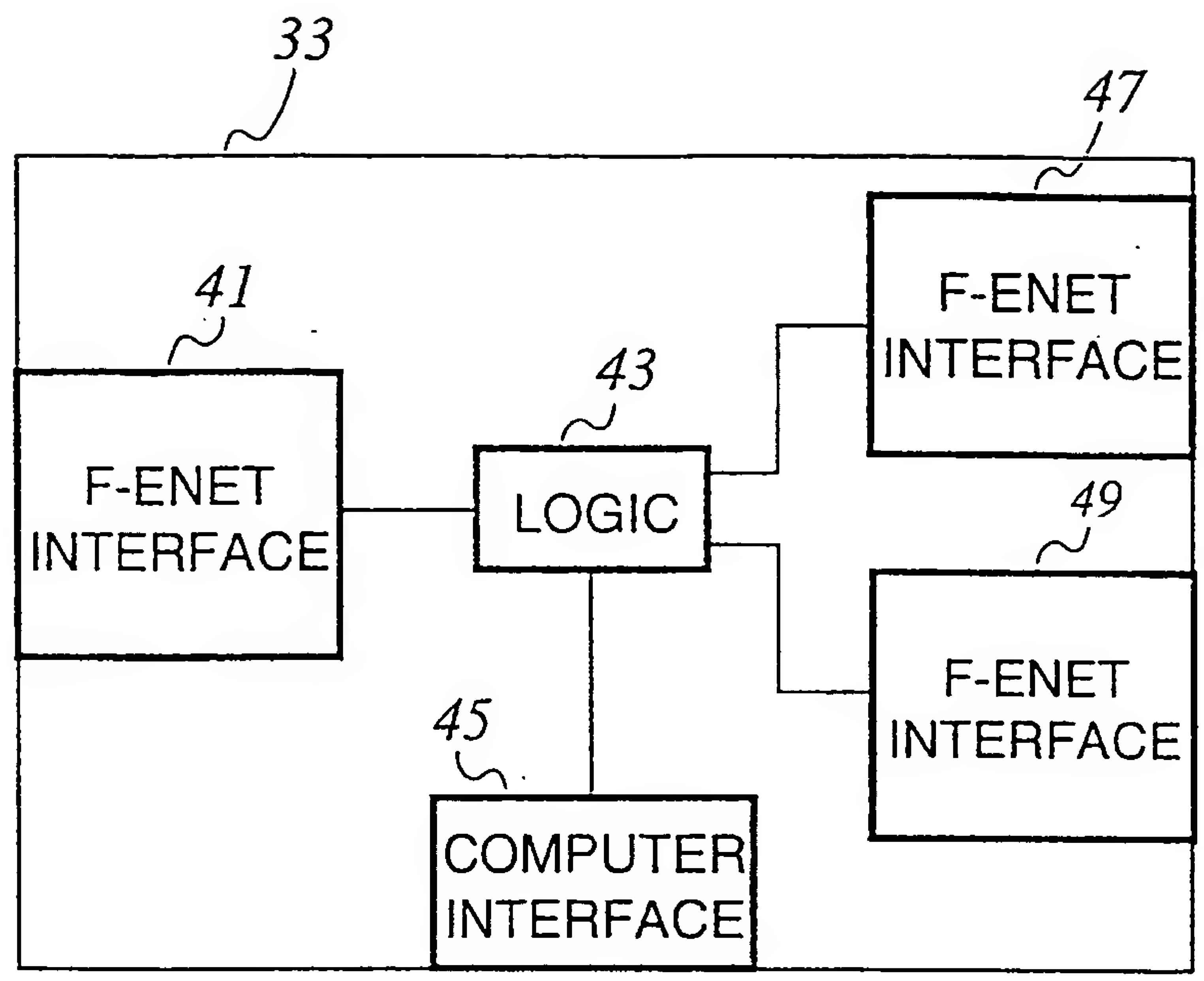

33
47
41
F-ENET
INTERFACE
43
LOGIC
49
F-ENET
INTERFACE
45
COMPUTER
INTERFACE
F-ENET
INTERFACE

FIG. 4

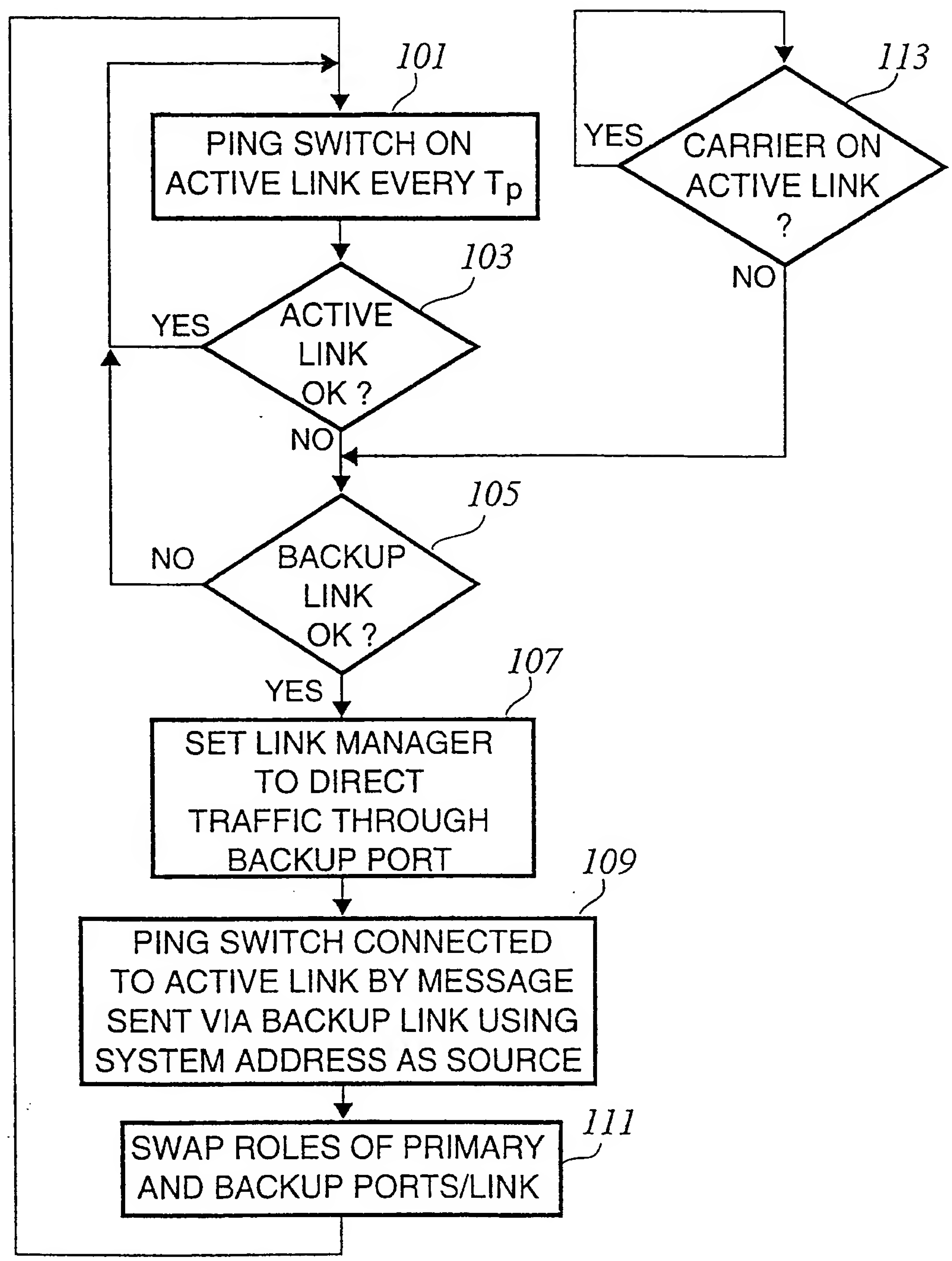

101
PING SWITCH ON ACTIVE LINK EVERY T_p
113
YES
CARRIER ON ACTIVE LINK ?
NO
103
YES
ACTIVE LINK OK ?
NO
105
NO
BACKUP LINK OK ?
YES
107
SET LINK MANAGER TO DIRECT TRAFFIC THROUGH BACKUP PORT
109
PING SWITCH CONNECTED TO ACTIVE LINK BY MESSAGE SENT VIA BACKUP LINK USING SYSTEM ADDRESS AS SOURCE
111
SWAP ROLES OF PRIMARY AND BACKUP PORTS/LINK

FIG. 5

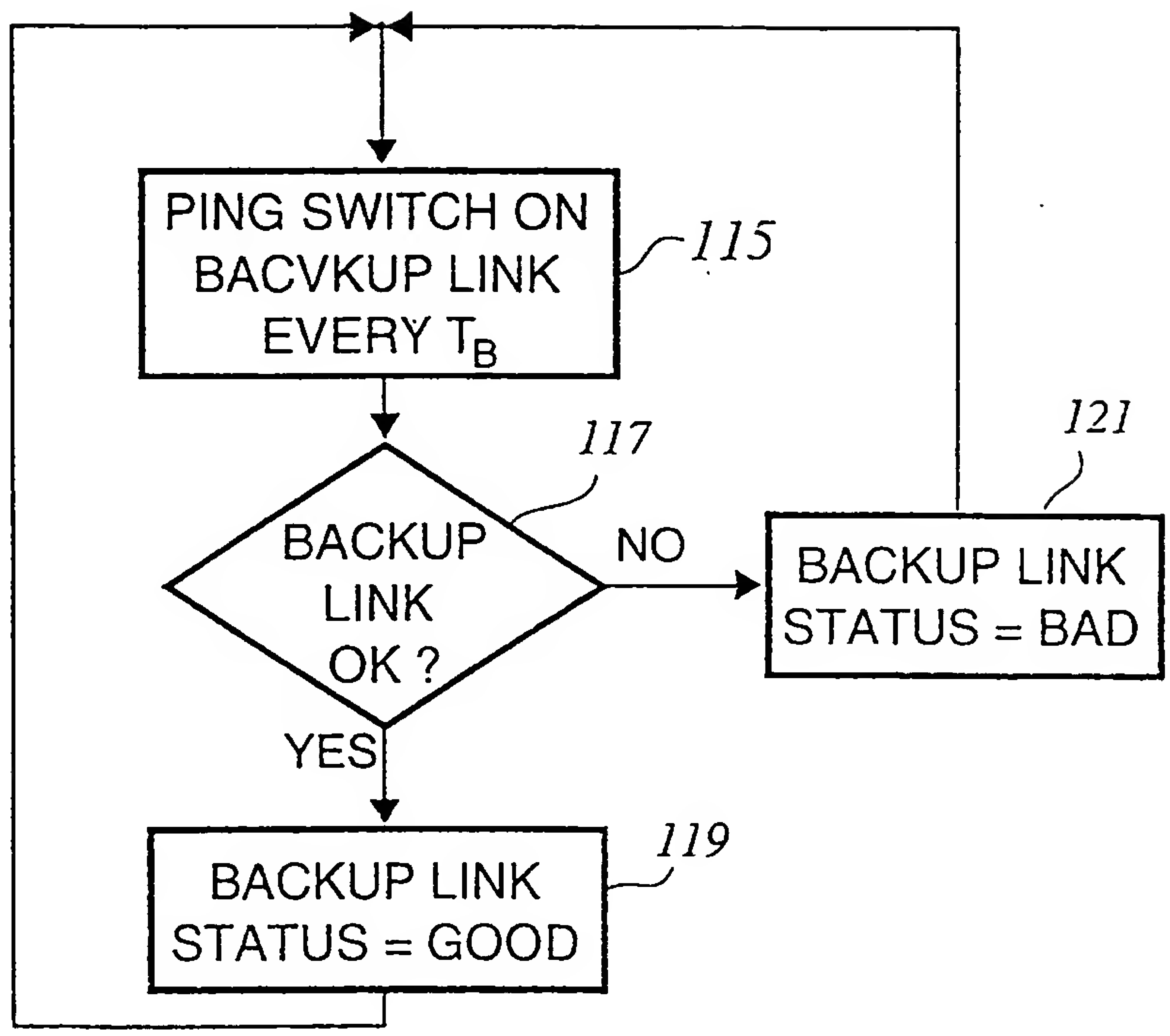
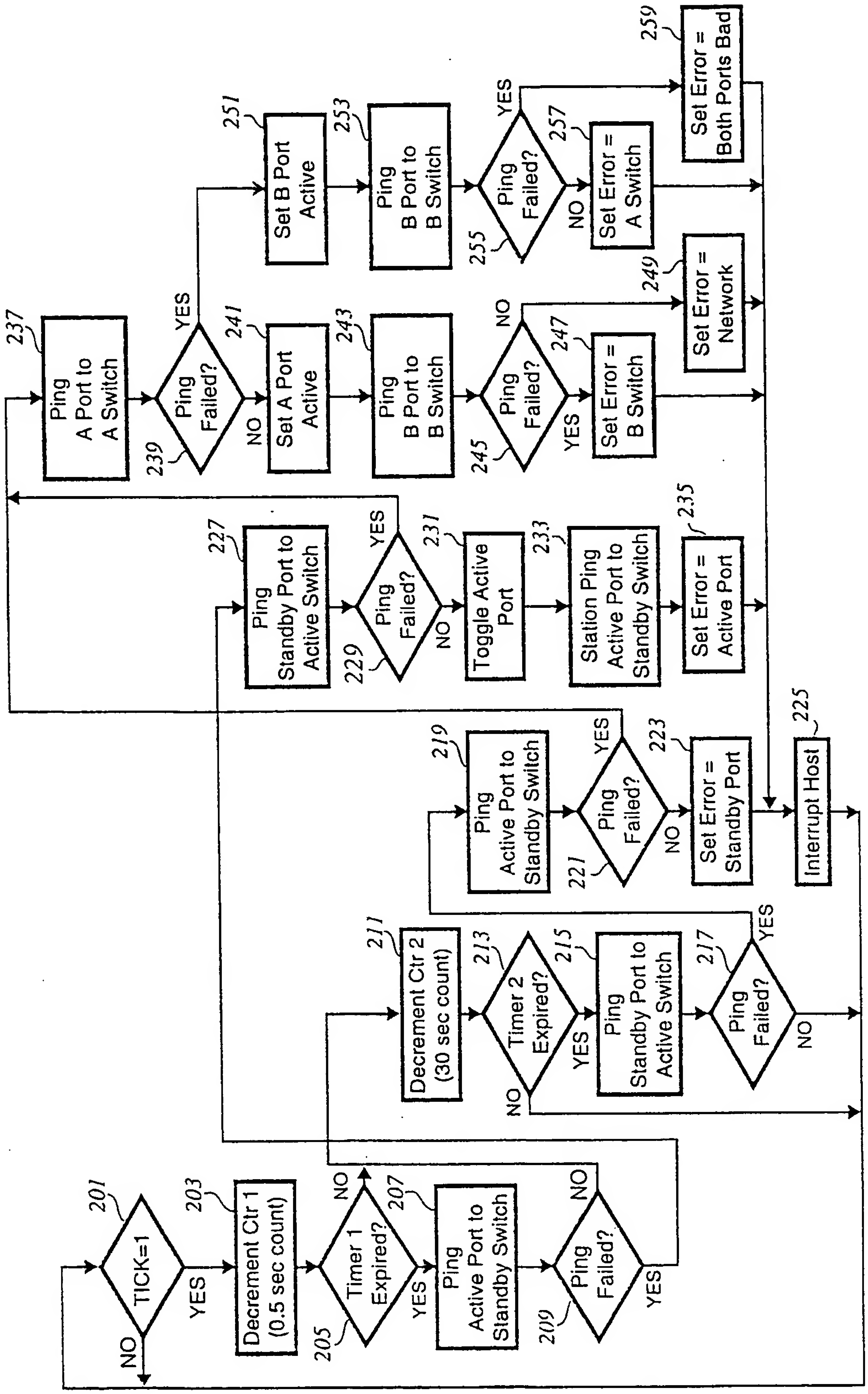

PING SWITCH ON BACVKUP LINK EVERY T_B
115
BACKUP LINK OK ?
117
NO
YES
BACKUP LINK STATUS = BAD
121
BACKUP LINK STATUS = GOOD
119

FIG. 6



201
TICK=1
YES
NO
203
Decrement Ctr 1 (0.5 sec count)
205
Timer 1 Expired?
YES
NO
207
Ping Active Port to Standby Switch
209
Ping Failed?
YES
NO
211
Decrement Ctr 2 (30 sec count)
213
Timer 2 Expired?
YES
NO
215
Ping Standby Port to Active Switch
217
Ping Failed?
YES
NO
219
Ping Active Port to Standby Switch
221
Ping Failed?
YES
NO
223
Set Error = Standby Port
225
Interrupt Host
227
Ping Standby Port to Active Switch
229
Ping Failed?
YES
NO
231
Toggle Active Port
233
Station Ping Active Port to Standby Switch
235
Set Error = Active Port
237
Ping A Port to A Switch
239
Ping Failed?
YES
NO
241
Set A Port Active
243
Ping B Port to B Switch
245
Ping Failed?
YES
NO
247
Set Error = B Switch
249
Set Error = Network
251
Set B Port Active
253
Ping B Port to B Switch
255
Ping Failed?
YES
NO
257
Set Error = A Switch
259
Set Error = Both Ports Bad

FIG. 7

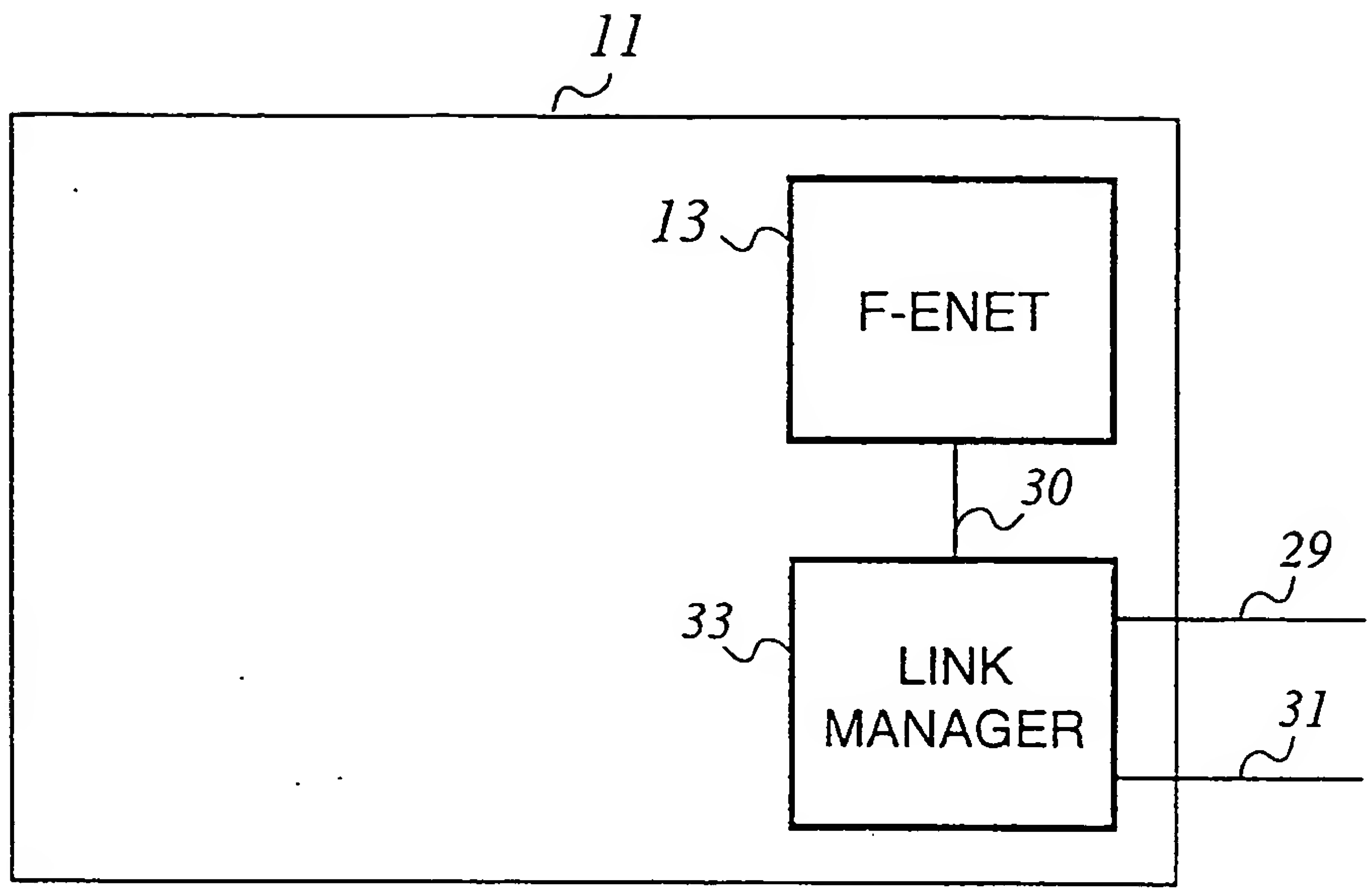

11
13
F-ENET
30
29
33
LINK
MANAGER
31

FIG. 8

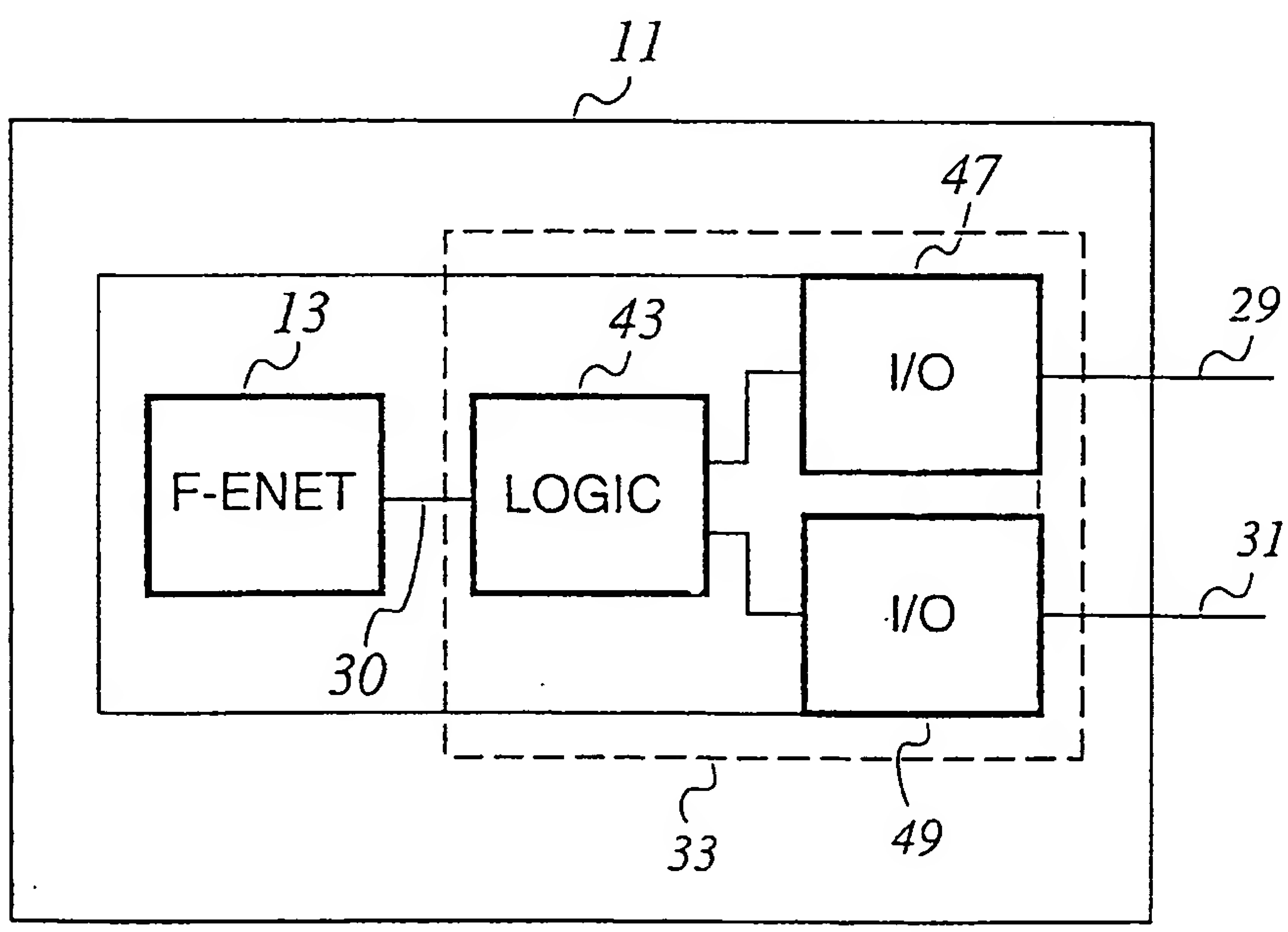

11
47
13
43
29
I/O
F-ENET
LOGIC
31
I/O
30
33
49

FIG. 9

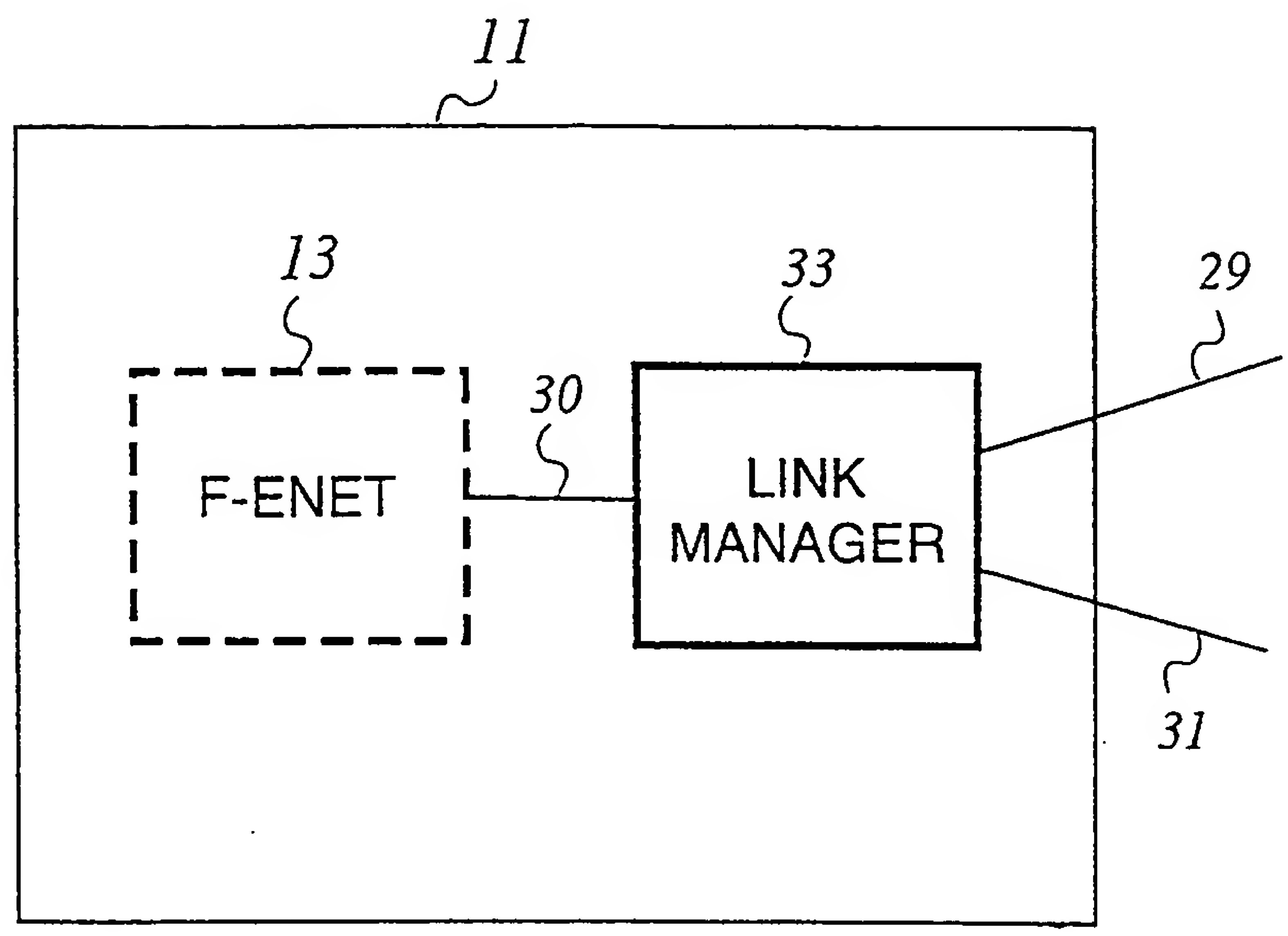

11
13
33
29
F-ENET
30
LINK
MANAGER
31

FIG. 10

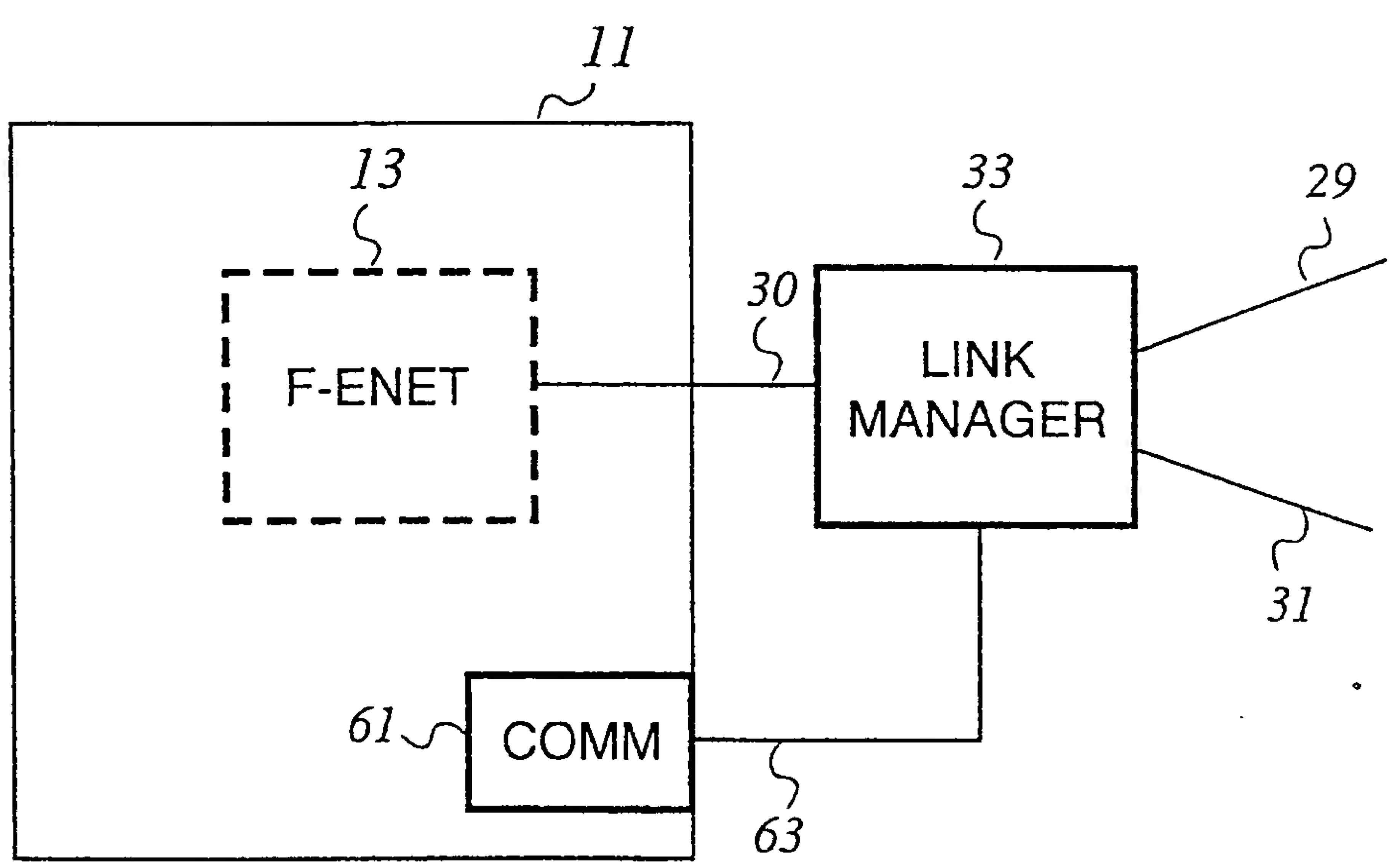

11
13
33
29
30
F-ENET
LINK
MANAGER
31
61
COMM
63

FIG. 11

INTERNATIONAL SEARCH REPORT

Inte. ..ional Application No
PCT/US 98/21984

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 H04L12/56 H04L29/14

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 4 692 918 A (ELLIOTT ROGER A ET AL) 8 September 1987 see column 1, line 60 - column 2, line 28 see column 12, line 63 - column 13, line 9	1-19
A	US 5 586 112 A (TABATA OSAMU) 17 December 1996 see abstract see column 3, line 47 - line 55 see claims 1-3	1,10,11
A	STEVENS ET AL: "TCP/IP ILLUSTRATED, Vol. 1. THE PROTOCOLS" TCP/IP ILLUSTRATED, VOL. 1: THE PROTOCOLS, vol. 1, pages 85-96, XP002106390 STEVENS;W R see the whole document	4-10

☐ Further documents are listed in the continuation of box C.

☒ Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
23 June 1999	06/07/1999

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Perez Perez, J

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/US 98/21984

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 4692918 A	08-09-1987	NONE	
US 5586112 A	17-12-1996	JP 2867860 B	10-03-1999
		JP 7177219 A	14-07-1995